On the Dilation of Point Sets

Rolf Klein, U Bonn
ADFOCS'05, MPI Saarbrücken

general problem statement:

given: set S of n points in the plane

wanted: "good" geometric network $N = (V, E)$ in $\mathbb{R}^2$ containing S

"good": low dilation, small number of edges,
low total weight (= sum of all edge lengths)
low degree / low diameter
few crossings
efficient construction algorithms
⋮

dilation: for vertices $p, q \in V$:

$\pi_N(p,q)$ a shortest path in N from p to q

aka stretch factor, spanning ratio, detour, distortion

$$\delta_N(p,q) := \frac{|\pi_N(p,q)|}{|pq|}$$

$|\pi_N(p,q)|$ ← Euclidean length

$|pq|$ ← Euclidean distance

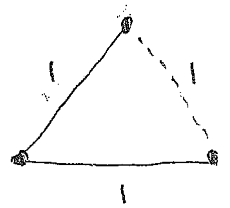
$$\delta(N) := \max_{\substack{p,q \in V \\ p \neq q}} \delta_N(p,q) \quad \text{dilation of } N$$

⇝ want good connections at low cost!

measures considered in this lecture:
dilation; number of edges, weight; construction time; crossings

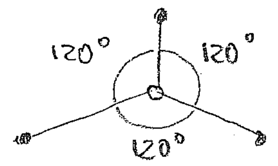
networks of lowest possible weight

- Euclidean minimum spanning tree $MST(S)$



here: $\delta = 2$, weight $= 2$

- Steiner tree, if using extra vertices is allowed



here: $\delta = \frac{2}{\sqrt{3}}$, weight $= \sqrt{3}$

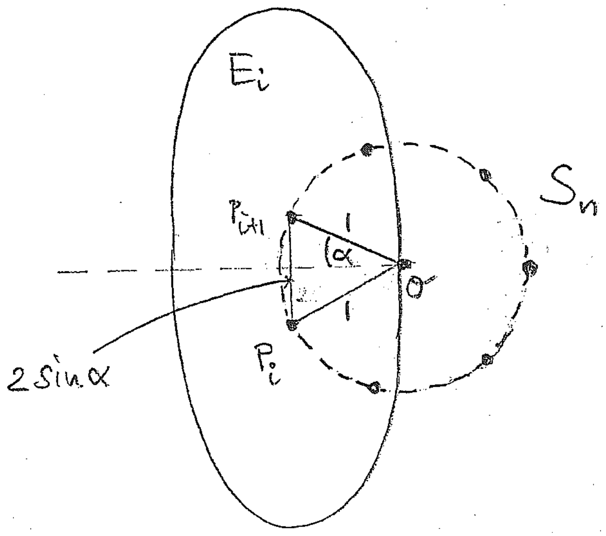
But: In general, trees can't give us low dilation

Proposition 1 Let $S_n :=$ vertex set of regular n-gon. Each tree $T = (V, E)$, where $S_n \subseteq V$, is of dilation

$$\delta(T) \geq \frac{1}{\pi} n .$$

Proof: Suppose $\delta(T) < \frac{n}{\pi} \leq \frac{1}{\sin \frac{\pi}{n}}$

since $\frac{\sin x}{x} \leq 1$



Let P_i, P_{i+1} neighbors in S_n
consider ellips E_i
with foci P_i, P_{i+1}
passing through o

$\Rightarrow$ E_i = locus of all points $z \in \Pi$
s.th. $|P_i z| + |z P_{i+1}| = 2$,

$|P_i P_{i+1}| = 2\sin\alpha$, $\alpha = \frac{\pi}{n}$

let v be a vertex of $\pi_T(P_i, P_{i+1})$

$$\Rightarrow \quad \frac{|P_i v| + v|P_{i+1}|}{2\sin\alpha} = \frac{|P_i v| + |v P_{i+1}|}{|P_i P_{i+1}|} \leq \delta(T) < \frac{1}{\sin\alpha}$$

$$\Rightarrow \quad |P_i v| + |v P_{i+1}| < 2 \quad \Rightarrow \quad v \in E_i$$

$$\Rightarrow \quad \pi_T(P_i, P_{i+1}) \subset E_i$$

same holds for every neighboring pair !

$\Rightarrow$ concatenation of shortest paths $\pi_T(P_i, P_{i+1})$
is a cycle in T that encircles origin o

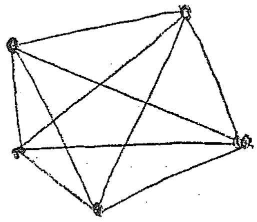
$\Rightarrow$ cycle not contractible

$\Rightarrow$ T is not a tree.

$\boxed{\text{Prop 1}}$

the other extreme : <u>network of lowest possible dilation</u>

complete graph over S



$\delta = 1$, but $\Theta(n^2)$ many edges
high weight, many crossings

<u>Question</u> Can we have dilation $1+\varepsilon$, $\Theta(n)$ many edges plus efficient construction?

Yes! Spanners.

(Source: G. Narasimhan, M. Smid: Geometric Spanner Networks Elsevier 2005 (hopefully))

[graph-theoretic algorithms] $\rightsquigarrow$ Uri Zwick's lecture

geometric algorithms

a little bit $\longrightarrow$ • Θ / Yao - graph

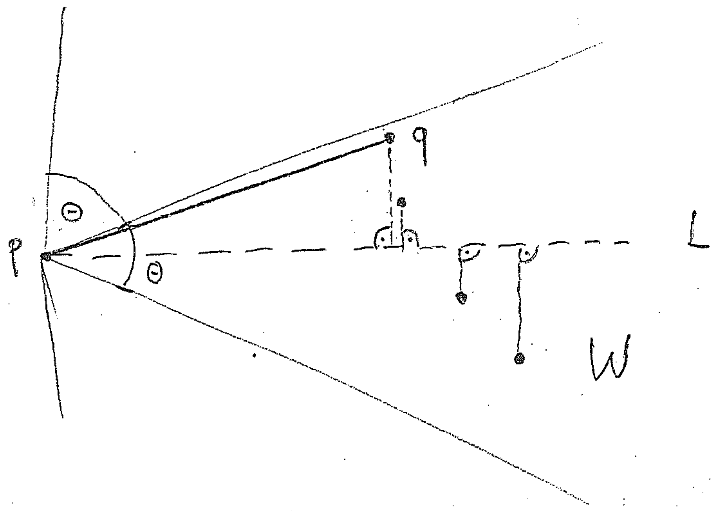
a little bit more $\longrightarrow$ • well-separated pair decomposition (WSPD)

<u>Θ-graph (in dimension 2)</u>

for each p in S

- partition plane into wedges of angle Θ around p
- choose halfline L in each wedge W
- determine point q in W
  - closest to p (Yao graph)
  - whose projection onto L is closest to p (Θ-graph)

- add edge (p,q) to spanner



Θ-graph has (i) $\frac{2\pi}{\Theta} n \in O(n)$ many edges

(ii) dilation $\frac{1}{\cos\Theta - \sin\Theta} \in O(1)$

(iii) can be constructed in time $O(\frac{1}{\Theta} n \log n) \subseteq O(n \log n)$

(Clarkson '87, Keil, Gutwin, Althöfer, Ruppert, Seidel; Yao, Chang, Huang, Tang)

Well-separated pair decomposition (WSPD)

(Callahan, Kosaraju '92–'95) → Narasimhan, Smid: Geometric Spanner Networks

Definition A pair of point sets A, B is well-separated with respect to s

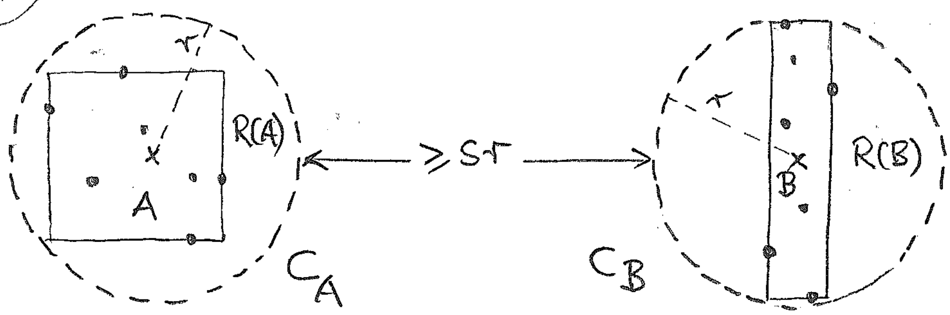
$:\Longleftrightarrow$ there are disks C_A, C_B of some radius r such that

- $C_A \cap C_B = \emptyset$
- C_A contains bounding box $R(A)$ of A; C_B " $R(B)$ of B
- $|C_A C_B| \geq s \cdot r$

bounding box makes decision possible in time $O(2^d)$

Vorschlag AG: verlange, das Kreiszentrum = Boxzentrum. Tut Beweis nicht weh

min distance between points of C_A, C_B



Usually, s is much larger than r = separation constant

Lemma 1 Let $a, a' \in A$ and $b, b' \in B$

(i) $|aa'| < 2r \leq 2\dfrac{|C_A C_B|}{s} \leq \dfrac{2}{s}|ab|$

(points on the same side are close, as compared to points on opposite sides)

(ii) $|a'b'| \leq |a'a| + |ab| + |bb'| \underset{(i)}{\leq} \left(1 + \frac{4}{s}\right)|ab|$

(all distances between points on opposite sides are almost equal)

Idea Represent given point set S as a finite union of well-separated pairs

Definition A well-separated pair decomposition of S for given parameter s is a sequence $(A_1, B_1), \dots, (A_m, B_m)$, where $A_i, B_i \subseteq S$,

such that
- A_i, B_i are well-separated w.r.t. s, $1 \le i \le m$
- for all $p \neq q$ in S there exists a unique i such that $p \in A_i$ and $q \in B_i$ or $q \in A_i$ and $p \in B_i$

Anwendung: Closest pair

do such things exist?

yes. we could simply use all singleton pairs $(\{a\}, \{b\})$

$\rightsquigarrow m = \Theta(n^2)$

can we do it with $m \in O(n)$ many pairs?

yes!

Theorem 1 Given a set S of n points in $\mathbb{R}^d$ and a parameter s, a WSPD of S with $m \in O((2s)^d d^{d/2} n)$ many pairs can be computed in time $O(dn \log n + (2s)^d d^{d/2} n)$

(s, d fixed: size $\in O(n)$, time $\in O(n \log n)$)

a property of Euclidean space

!

doesn't mean $\sum_{i=1}^{m} |A_i| + |B_i| \in O(n)$!

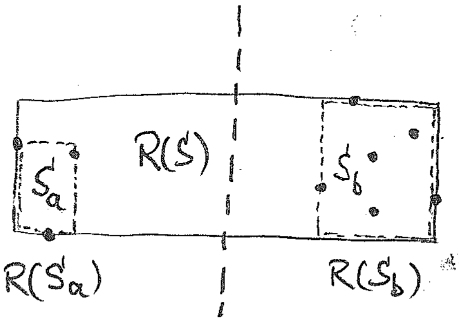
$\rightarrow$ E3

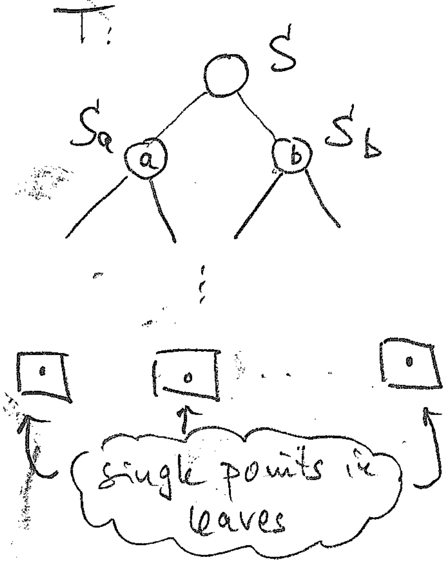
Proof: ① algorithm (extremely simple)

anders als beim kd-Baum

start with bounding box $R(S)$;
halve longest dimension of $R(S)$ by orthogonal hyperplane
and recur on resulting subsets S_a, S_b of S
until singletons left

not necessarily balanced





$\rightsquigarrow$ split tree T

shows recursion history

- does not depend on parameter s
- nodes $a \longleftrightarrow$ certain subsets S_a of S
- can be constructed in $O(dn\log n)$ time

verzeigerte Listen verwenden

- passing down sorted lists of coordinates for each dimension (for bounding box computation) (as in range trees)
- splitting lists in time ~ smaller subset *)

one phase {
- by recursively constructing partial split trees containing $\leq \frac{n}{2}$ points in each leaf

to obtain partial split tree of $\frac{n}{2}$ points in each leaf: spalte immer – nach obiger Methode – die anzahlmäßig größere verbleibende Menge. (die andere ist $< \frac{n}{2}$) Das geht insgesamt in Zeit $O(n)$ für diese erste Phase

*) two pointers moving in from either end, until split value found

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} = \underbrace{\frac{1}{k!}}_{\text{Konst.}} \overbrace{(n-k+1)(n-k+2)\cdots(n)}^{k \text{ Faktoren} \le n} \le \frac{1}{k!} n^k$$

8.1

$$\vee\!\mid$$

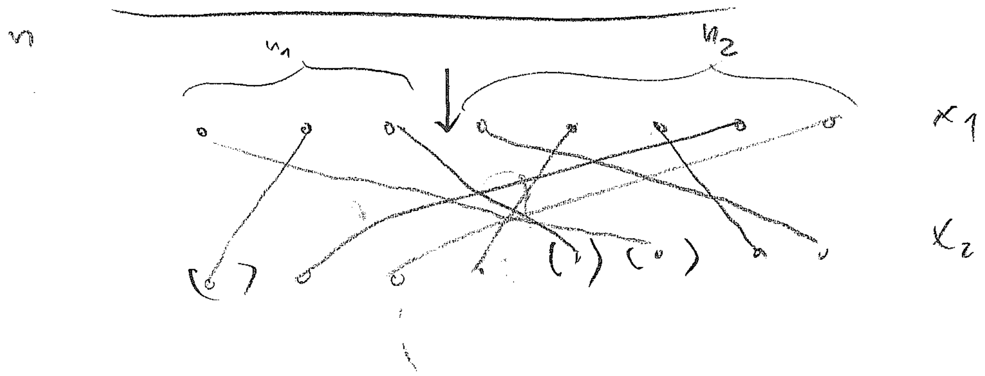
$$\frac{1}{k!}(n-k+1)^k$$

$$(n-k+1)^k = n^k \left(\frac{n-k+1}{n}\right)^k = n^k \underbrace{\left(1-\frac{k-1}{n}\right)^k}$$

$$\vee\!\mid$$

$\left(\frac{1}{2}\right)^k$ für n groß

$n_1 \le n_2 \Rightarrow n_1 \le \frac{n}{2}$

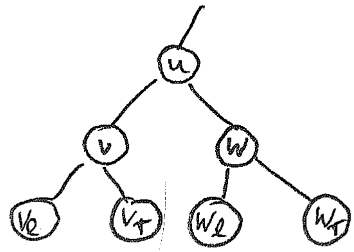


Liste Länge n Splitten in $n_1 \le \frac{n}{2}$, $n_2 \ge \frac{n}{2}$: $d n_1$ ...

to obtain WSPD of S w.r.t. s:

for each internal node u of T with children v, w

invoke procedure



using bounding boxes: $O(d)$

FindPairs (v, w):

if S_v, S_w are well-separated w.r.t. s then report (v, w)

else if $L_{max}(R(S_v)) > L_{max}(R(S_w))$

longest dimension

then FindPairs(v_ℓ, w), FindPairs(v_r, w)

else FindPairs(v, w_ℓ), FindPairs(v, w_r)

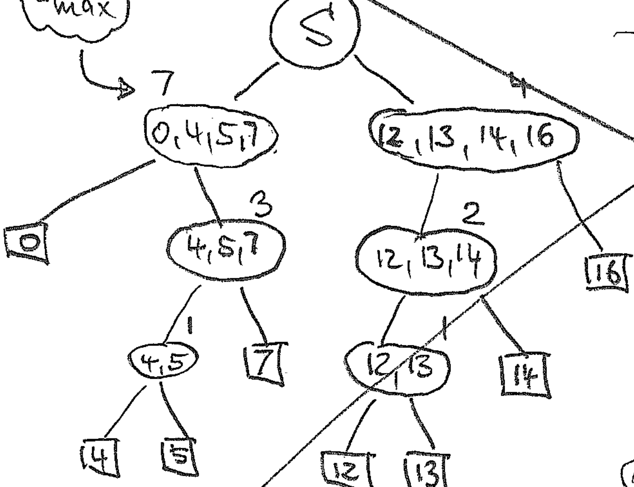
clear: algorithm terminates and returns WSPD

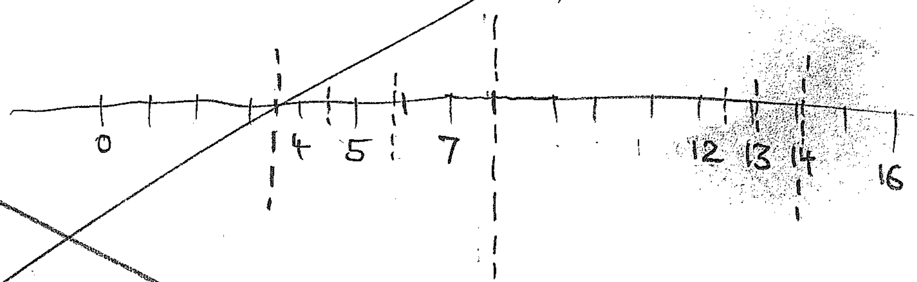
not clear: number of pairs, m, reported

existence of singletons → 9.1

(there are only $O(n)$ many subsets S_a, corresponding to the nodes of T, but the same set can occur many times as A_i, A_j, B_k etc.)

Example $d = 1$, $S = \{0, 4, 5, 7, 12, 13, 14, 16\}$; $s = 2$





WSPD:

S: $(\{0\}, \{12, 13, 14, 16\})$, $(\{4,5,7\}, \{16\})$, $(\{4,5\}, \{12,13,14\})$, $(\{7\}, \{12,13,14\})$;

0,4,5,7: $(\{0\}, \{4,5\})$, $(\{0\}, \{7\})$; 4,5,7: $(\{4,5\}, \{7\})$; 4,5: $(\{4\}, \{5\})$

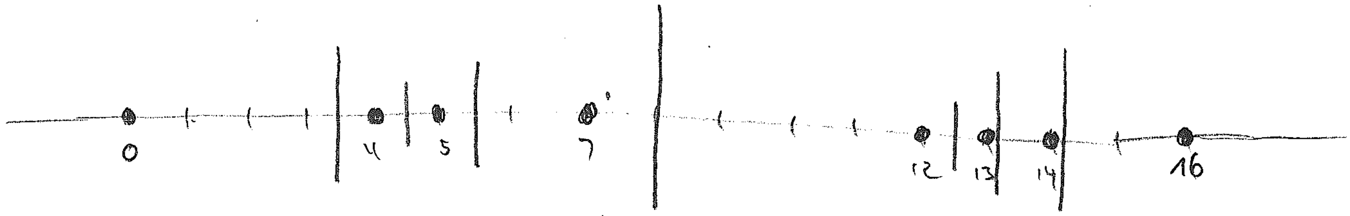
12,13,14,16: $(\{12,13\}, \{16\})$, $(\{14\}, \{16\})$; 12,13,14: $(\{12\}, \{14\})$, $(\{13\}, \{14\})$

12,13: $(\{12\}, \{13\})$.

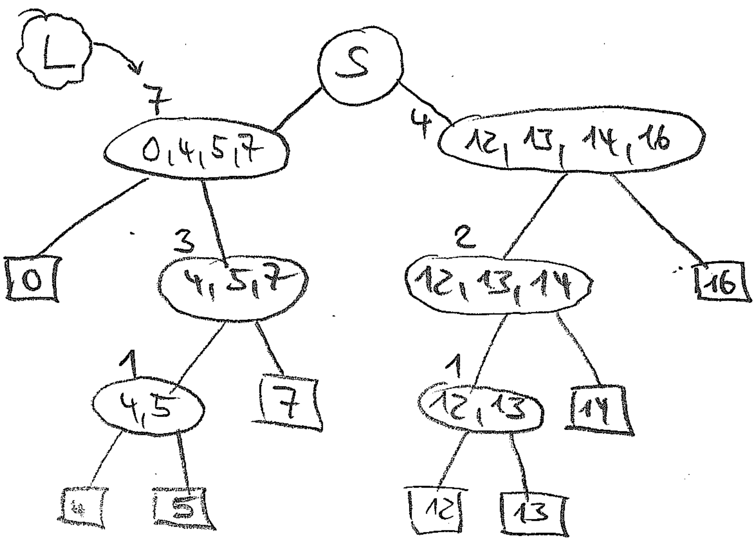
subset $\{4,5\}$ occurs 3 times

Beispiel für eine WSPD

$d = 1$, $S = \{0, 4, 5, 7, 12, 13, 14, 16\}$, $s = 3$



Split Tree $T(S)$

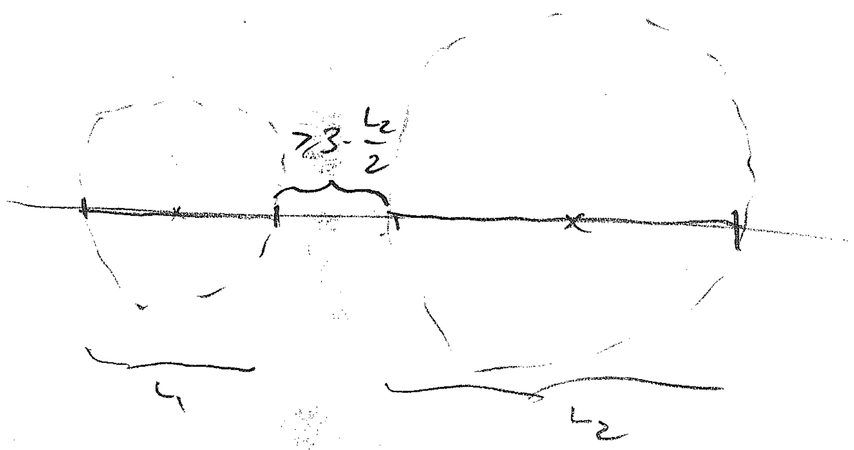


In Dimension 1: L_{max} = Länge L der Menge

$\frac{L}{2}$ = Radius des kleinsten diese Menge einschließenden "Kreises"

Also: A, B wohlsepariert bezüglich $s = 3$

$\Longleftrightarrow$ Abstand $\geq \frac{3}{2}$ maximale Länge



$L_1 < L_2$

Wald der Rekursionsaufrufe von FP = FindPairs (die Mengen in den Blättern sind wohlsepariert und werden berichtet):

- FP({0,4,5,7}, {12,13,14,16})
  - FP({0}, {12,13,14,16})
  - FP({4,5,7}, {12,13,14,16})
    - FP({4,5,7}, {12,13,14})
    - FP({4,5,7}, {16})

- FP({0}, {4,5,7})
  - FP({0}, {4,5})
  - FP({0},{7})

- FP({12,13,14}, {16})
  - FP({12,13}, {16})
  - FP({14}, {16})

- FP({4,5}, {7})

- FP({12,13}, {14})
  - FP({12},{14})
  - FP({13}, {14})

- FP({4},{5})

- FP({12}, {13})

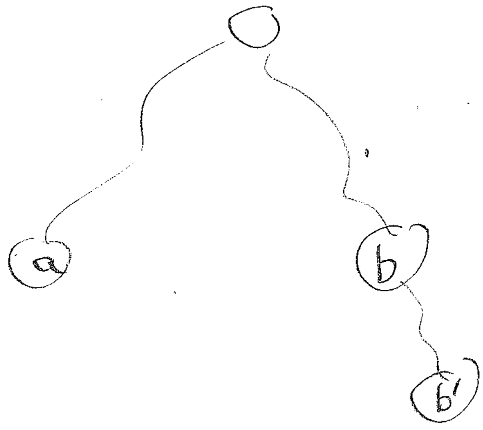
WSPD (in lexikographischer Reihenfolge)

({0}, {4,5}), ({0}, {7}), ({0}, {12,13,14,16}), ({4}, {5}), ({4,5}, {7}), ({4,5,7}, {12,13,14}), ({4,5,7}, {16}), ({12}, {13}), ({12}, {14}), ({12,13}, {16}), ({13}, {14}), ({14}, {16}).

Some sets occur more than once.

9.1

Eindeutigkeitseigenschaft der WSPD: Folgt daraus, daß niemals

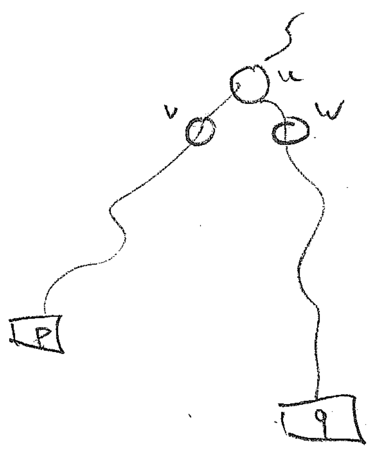


(a,b), (a,b') berichtet

(sondern b,b' immer durch Hyperebene getrennt)

Wie oft kann das vorkommen?

Existenz: Sei $p \neq q$; starte rückwärts bei den Blättern



Terminiert wegen Singletons

Lemma i, j Knoten von T $\Longrightarrow$
$i \neq j$

$S_i \subset S_j$ oder $S_j \subset S_i$ oder $S_i \cap S_j = \emptyset$

Question: how many pairs (a, b_i) can occur?
Clear : $S_{b_i} \cap S_{b_j} = \emptyset$!

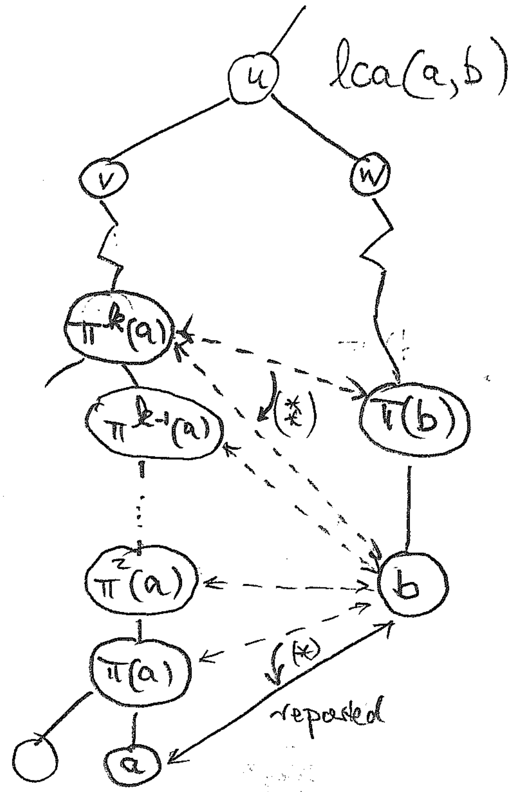
(2) analysis

suppose (a,b) gets reported, since S_a, S_b well-separated

$\Rightarrow$ $S_{\pi(a)}, S_b$ or $S_a, S_{\pi(b)}$ are NOT well-separated,

focus on this type

where $\pi(a)$ = parent node of a



{ tests made by FindPairs(v,w) v,w sons of u

imply :

$$L_{max}(R(b)) \leq L_{max}(R(\pi(a))) \qquad (*)$$

$$L_{max}(R(\pi(a))) \leq L_{max}(R(\pi^k(a)))$$

Vater immer größer

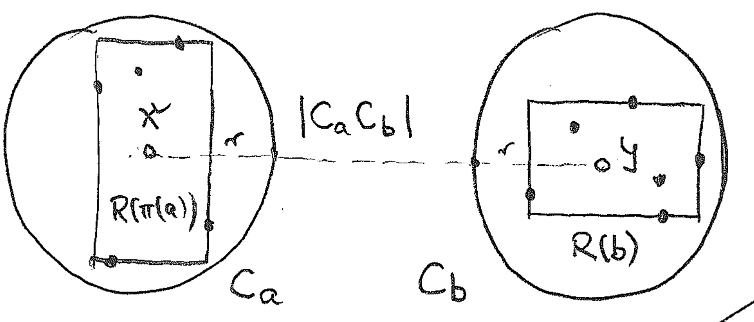
$$\leq L_{max}(R(\pi(b))) \qquad (**)$$

if $\pi(b) = u$, (**) trivially holds

draw circles of radius $\frac{\sqrt{d}}{2} L_{max}(R(\pi(a))) =: r$

around center $=: x$ of $R(\pi(a))$

$=: y$ of $R(b)$



R(b) fits in C_b because of (*)
(2)

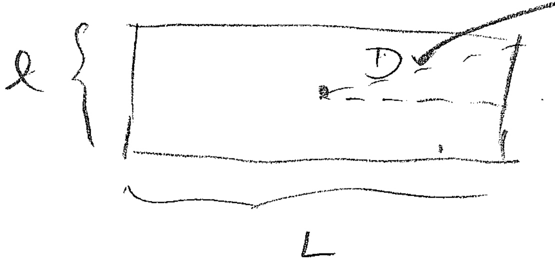
$\frac{\sqrt{d}}{2}$ = length from center of d-box to furthest ~~vertex~~ length of longest edge
(1a)

$R(\pi(a))$ fits in C_a because of $\frac{\sqrt{d}}{2}$ factor
(1)

Proof of 1a for $d=2$

$d=2$

$D \overset{!}{\leq} \frac{\sqrt{2}}{2} \cdot L$



$$D = \sqrt{\left(\frac{L}{2}\right)^2 + \left(\frac{\ell}{2}\right)^2} = \frac{1}{2}\sqrt{L^2+\ell^2}$$

$$\leq \frac{1}{2}\sqrt{2L^2}$$

$$= \frac{\sqrt{2}}{2} L.$$

if $C_a \cap C_b = \phi$

$\Longrightarrow \quad |C_a C_b| < s \cdot r \quad \Rightarrow \quad |xy| = |C_a C_b| + 2r < \left(\frac{s}{2}+1\right) 2r$

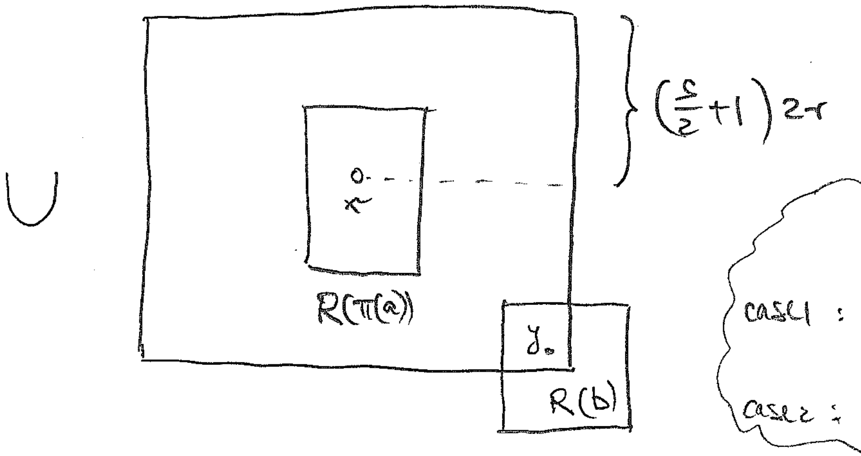
$S_{\pi(a)}, S_b$ not well-sep

boxes $R(\pi(a)), R(b)$ are close

if $C_a \cap C_b \neq \phi$

$\Longrightarrow \quad |xy| \leq 2r < \left(\frac{s}{2}+1\right) 2r$

$\Longrightarrow$ y contained in hypercube U of size $2\left(\frac{s}{2}+1\right)2r$ centered at x



tree property:
case1: a predecessor of b (or vice versa)
$\Rightarrow S_a \supseteq S_b$
case2: $S_a \cap S_b = \phi$

now assume that pairs $(a, b_1), \ldots, (a, b_k)$ of this type are reported in WSPD

$\longrightarrow$ 11.1

$\Longrightarrow$ for $i \neq j$: S_{b_i}, S_{b_j} disjoint, hence seperated by hyperplanes

uniqueness of WSPD, S_a always involved

$\Longrightarrow R(b_i), R(b_j)$ disjoint

want to bound $\underset{\#\text{pairs}}{\overset{}{k}}$ by packing argument

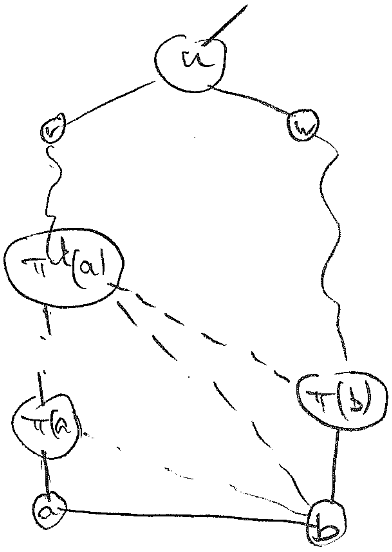
need lower bound to volume of $R(b_i)$

?

Rekapitulation

FindPairs(v, w) berichtet (a, b) als wohlsep., aber (π(a), b) nicht

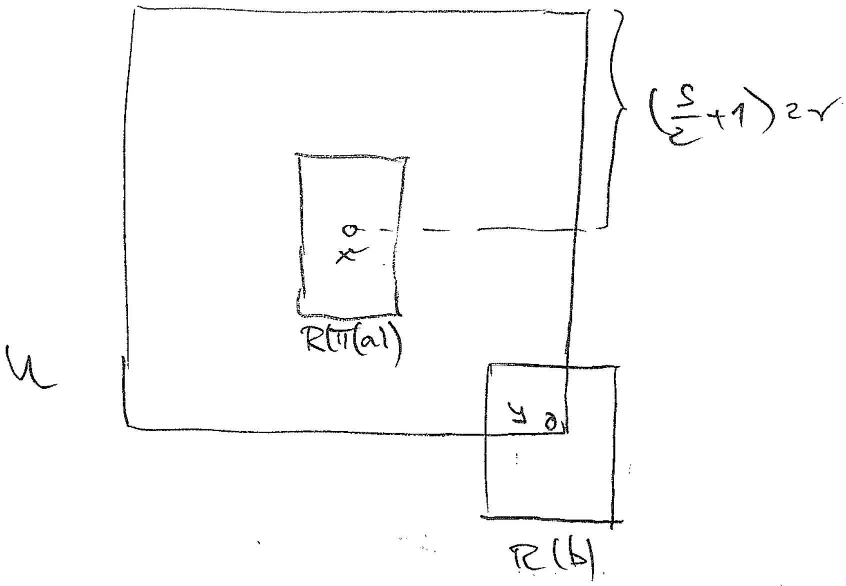
11.0



$$L_{max}(R(\pi(a))) \leq L_{max}(R(\pi(b))) \quad (*)$$

$$r := \frac{\sqrt{d}}{2} L_{max}(R(\pi(a)))$$

Hatten gesehen: Bounding Boxes von $R(\pi(a))$, $R(b)$ "nah beisammen"



24 x 50.000

1.200.000

11.1

Es gibt immer zwei Möglichkeiten für die Genese von (a,b):

$S_{\pi(a)}, S_b$ nicht well-separated $\Rightarrow$ write (a,b)

oder $S_a, S_{\pi(b)}$ nicht well-separated $\Rightarrow$ write (b,a)

} $\Rightarrow$ we know: all $S_{\pi(a)}, S_{b_i}$ not well-separated

~~Beweisen, daß Eindeutigkeit erfüllt:~~

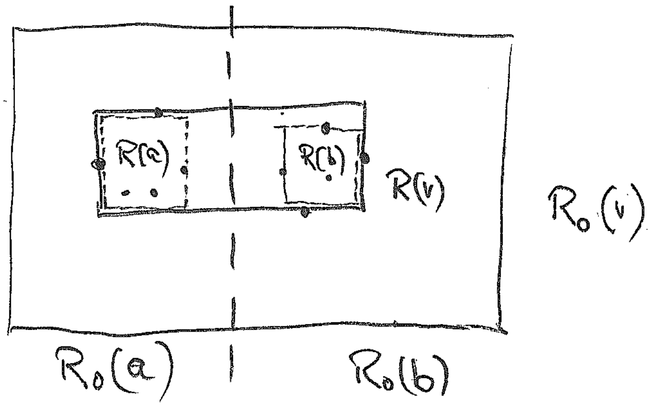
~~S_a, S_b berichtet~~

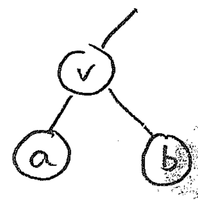
interlude : just for the proof

maintain container $R_0(a)$ for bounding box $R(a)$,
for each set S_a of the split tree

start with hyper**cube** $R_0(S)$ containing $R(S)$
when splitting a bounding box, use same hyperplane for splitting the container :

containers don't shrink to fit point set





clear: $S_v \cap S_w = \emptyset \Rightarrow R_0(v) \cap R_0(w) = \emptyset$

Claim $b \neq \text{root} \Rightarrow L_{min}(R_0(b)) \overset{!}{\geq} \frac{1}{2} L_{max}(R(\pi(b)))$

Proof by induction.

$\pi(b) = \text{root} \Rightarrow L_{min}(R_0(b)) = \frac{1}{2}$ size of cube $R_0(S)$

$= \frac{1}{2} L_{max}(R(S))$ ✓

$\pi(b) \neq \text{root}$

case 1 : $L_{min}(R_0(b)) = L_{min}(R_0(\pi(b))) \underset{\text{induct.}}{\geq} \frac{1}{2} L_{max}(R(\pi^2(b)))$

$\geq \frac{1}{2} L_{max}(R(\pi(b)))$ ✓

case 2 : $L_{min}(R_0(b)) < L_{min}(R_0(\pi(b)))$

$\Rightarrow$ new min dimension = split dimension, i

13.0

$\Rightarrow \quad L_{min}(R_0(b)) = L_i(R_0(b)) \geq$ [illegible] $= \frac{1}{2} L_i(R(\pi(b)))$

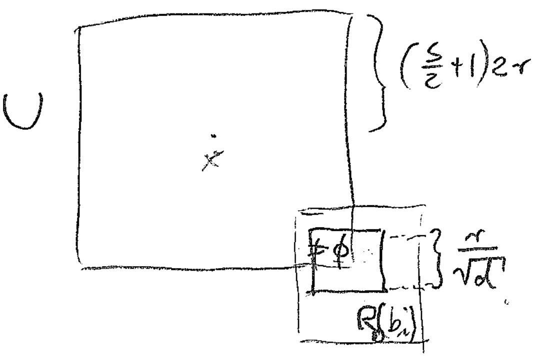
$= \frac{1}{2} L_{max}(R(\pi(b)))$ ✓

by definition of split algorithm

$\boxed{\text{Claim}}$

proof continued

Erinnerung: $R(b_i) \cap U \neq \emptyset$, $R(b_i) \subset R_0(b_i)$



Know for $b_1, \dots, b_k$

- $R_0(b_i) \cap R_0(b_j) = \emptyset$
- $L_{max}(R(\pi(a))) \underset{(*)}{\leq} L_{max}(R(\pi(b_i))) \underset{\text{claim}}{\leq} 2 \cdot L_{min}(R_0(b_i))$

can flesh up $R(b_i)$ now

$\Rightarrow$ each $R_0(b_i)$ contains hypercube of size

$\frac{1}{2} L_{max}(R(\pi(a))) = \frac{r}{\sqrt{d}}$ that intersects U w.l.o.g.

def-1

wissen: $\emptyset \neq U \cap R(b_i) \subseteq U \cap R_0(b_i)$

volume $= \left(\frac{r}{\sqrt{d}}\right)^d$

move such hypercube within $R_0(b_i)$ until it intersects U

$\Rightarrow$ union of U and all (these) hypercubes contained in supercube of size $2(\frac{s}{2}+1)2r + \frac{2r}{\sqrt{d}} \leq (2s+6)r$

volume $\leq (2s+6)^d r^d$

$\Rightarrow$ hypercubes disjoint, vol $\left(\frac{r}{\sqrt{d}}\right)^d$ each

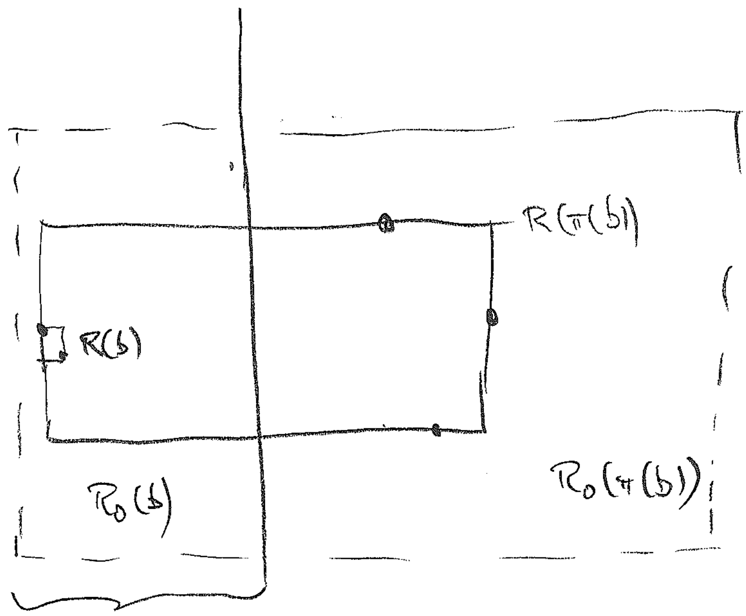
$k = \#\text{hypercubes} \leq (2s+6)^d \sqrt{d}^d \in O(2^d s^d d^{d/2})$

$=$ # Paare (a, b_i) reported

$\Rightarrow$ total number of pairs $\in O(2^d s^d d^{d/2} n)$

Observe: $\sum_{i=1}^{m} (|A_i| + |B_i|)$ can be $\sim n^2$

But (A_i, B_i) is represented by 1 pointer



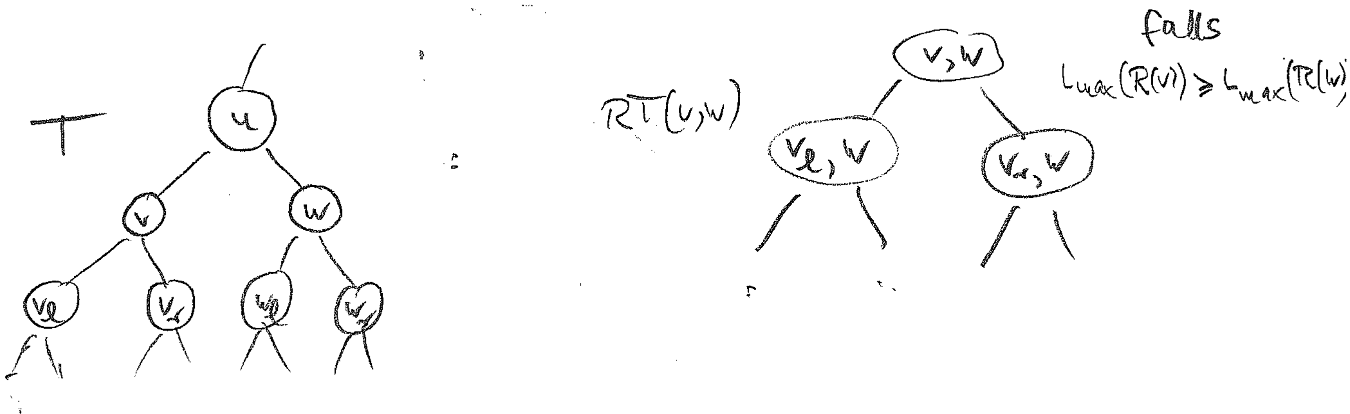
$$L_i(R_0(b)) \geq \frac{1}{2} L_i(R(\pi(b))) = \frac{1}{2} L_{max}(R(\pi(b)))$$

Def.

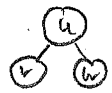
Laufzeit des Algorithmus:

- Jedes interne Knoten u von T gibt Anlaß zu einem Rekursionsbaum $RT(v,w)$

mit Kindern v,w



- Der Kopf jedes FindPairs-Aufruf kostet $O(2^d)$ Zeit (Test, ob S_v, S_w wohlsepariert, berechne die L_{max}-Werte)

- Paare der WSPD $\longleftrightarrow$ sämtliche Blätter der Rekursionsbäume $RT(v,w)$, wo  in T, d.h. v,w Geschwister

- Gesamtzahl aller FindPairs-Aufrufe
  $=$ Gesamtzahl der Knoten aller Rekursionsbäume $RT(v,w)$
  $\in O($Gesamtzahl aller Blätter$) = O(2^d s^d d^{\frac{d}{2}} n)$

- Gesamtkosten in $O(2^d s^d d^{\frac{d}{2}} n)$, um WSPD aus SplitTree zu berechnen

$O(d n \log n)$

Damit Theorem 1 bewiesen

Theorem 2 Sei $S \subseteq \mathbb{R}^d$, $|S| = n$, $\varepsilon > 0$.

Dann kann man einen $(1+\varepsilon)$-Spanner N von S mit $O(\frac{1}{\varepsilon^d} n)$ Kanten in Zeit $O(\frac{1}{\varepsilon^d} n + d n \log n)$ berechnen.

D.h. 1) für je zwei Punkte $p, q \in S$ gibt es in N einen Pfad der Länge $\leq (1+\varepsilon) |pq|$ von p nach q

2) N enthält nur $O(\frac{1}{\varepsilon^d} n)$ Kanten

(14.6.3)

measures considered in this lecture:

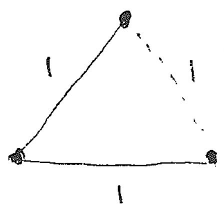
dilation; number of edges, weight; construction time; crossings

networks of lowest possible weight

- Euclidean minimum spanning tree MST(S)

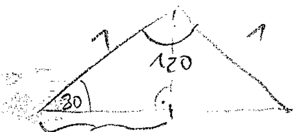
Dillis

aber hohe Dilation

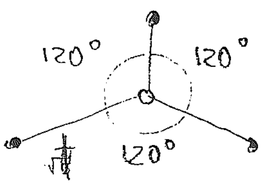


here: $\delta = 2$, weight $= 2$

- Steiner tree, if using extra vertices is allowed (they also write)



$\cos 30° = \frac{\sqrt{3}}{2}$



here: $\delta = \frac{2}{\sqrt{3}}$, weight $= \sqrt{3}$

But: In general, trees can't give us low dilation

Proposition 1 Let $S_n :=$ vertex set of regular n-gon.

Each tree $T = (V, E)$, where $S_n \subseteq V$, is of dilation

$$\delta(T) \geq \frac{1}{\pi} n$$

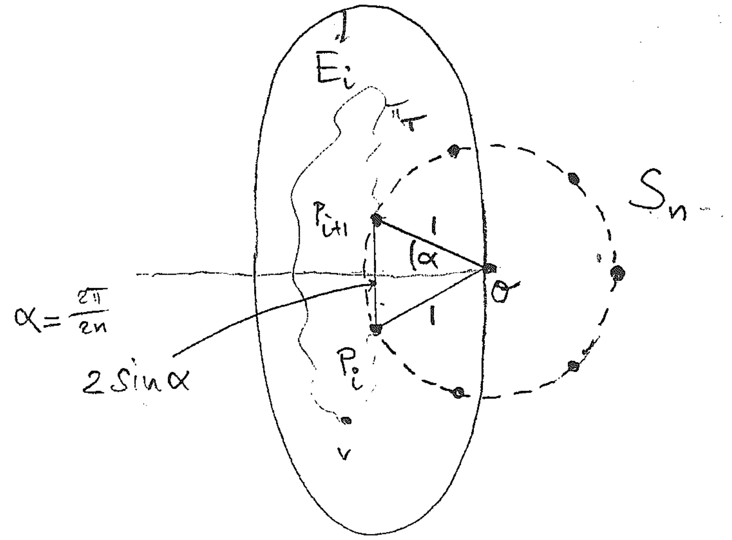
(between the points of S_n)

Steinerpunkte erlaubt

Proof: Suppose $\delta(T) < \frac{n}{\pi}$ $\left[\leq \frac{1}{\sin \frac{\pi}{n}} \right.$

since $\frac{\sin x}{x} \leq 1$ $\left. \right]$

erst auf der nächsten Seite



Let p_i, p_{i+1} neighbors in S_n
consider ellipse E_i
with foci p_i, p_{i+1}
passing through o

$\Rightarrow$ E_i = locus of all points $z \in \pi$
s.th. $|p_i z| + |z p_{i+1}| \leq 2$,

let v be a vertex of $\pi_T(p_i, p_{i+1})$

$$\Rightarrow \quad \frac{|p_i v| + |v p_{i+1}|}{2\sin\alpha} = \frac{|p_i v| + |v p_{i+1}|}{|p_i p_{i+1}|} \leq \delta(T) < \frac{1}{\sin\alpha}$$

$\leq \frac{|\pi_T|}{|p_i p_{i+1}|}$

$\frac{n}{\pi} <$ $\frac{1}{\sin\frac{\pi}{n}} = \frac{1}{\sin\alpha}$

$$\Rightarrow \quad |p_i v| + |v p_{i+1}| < 2 \quad \Rightarrow \quad v \in \mathring{E}_i$$

$$\Rightarrow \quad \pi_T(p_i, p_{i+1}) \subset \mathring{E}_i$$

same holds for every neighboring pair !

$\Rightarrow$ concatenation of shortest paths $\pi_T(p_i, p_{i+1})$ is a cycle in T that encircles origin o

$\Rightarrow$ cycle not contractible

$\Rightarrow$ T is not a tree.

$\boxed{\text{Prop 1}}$

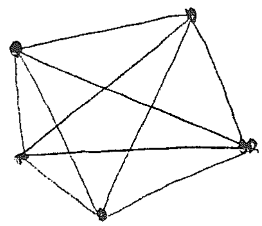
for general graphs:
- finding min dilation tree is NP-hard (Cai, Corneil '95)
- $\log n$ approximation (Emek, Peleg 04)

also for points in $\mathbb{R}^2$

the other extreme : network of lowest possible dilation

low dilation, very expensive

complete graph over S



$\delta = 1$, but $\Theta(n^2)$ many edges
high weight, many crossings

Question Can we have dilation $1+\varepsilon$, $O(n)$ many edges plus efficient construction?

Yes! Spanners.

(Source: G. Narasimhan, M. Smid: Geometric Spanner Networks
Elsevier 2006 (hopefully))

[graph-theoretic algorithms] $\rightsquigarrow$ Uri Zwick's lecture

geometric algorithms

a little bit $\longrightarrow$ • Θ / Yao-graph

a little bit more $\longrightarrow$ • well-separated pair decomposition (WSPD)

Θ-graph (in dimension 2)

for each p in S

- partition plane into wedges of angle Θ around p
- choose halfline L in each wedge W
- determine point q in W
  - closest to p (Yao graph)
  - or – whose projection onto L is closest to p (Θ-graph)

applications of WSPD

erst die übrigen Anwendungen

~~2.)~~ spanner construction: d fixed

let $s>4$, take one edge from each pair (A_i, B_i) → graph G

$\xrightarrow[\text{Lemma 1}]{\ddot{u}}$ $\frac{s+4}{s-4}$-spanner with $O(n)$ edges in time $O(n\log n)$

hint: [induction on rank of distance $|pq|$]

2) detailed proofs of Theorem 2: consider sorted sequence of distances $|pq|$.

by induction on rank: there is a $t := \frac{s+4}{s-4}$ path in G connecting p, q

~~$|pq| = 0$~~ ✓ : (p,q closest pair ⇒ singleton ⇒ edge was picked)

singletons, if $|pq| > 0$ minimal.

$|pq| > 0$ Let $p \in A_i$, $q \in B_i$, (A_i, B_i) pair of WSPD

let $p' \in A_i$, $q' \in B_i$ be the pair chosen for G

Then $|pp'| \underset{\text{Lemma 1, (i)}}{\leq} \frac{2}{s}|pq| \underset{s>4}{<} |pq| \underset{\text{induct.}}{\Longrightarrow}$ ex t-path $\pi_{pp'}$ in G

also, ex t-path $\pi_{q'q}$ in G. Moreover, $|p'q'| \leq (1+\frac{4}{s})|pq|$ by Lemma 1, (ii).

Consider $\pi_{q'q} \circ \overline{p'q'} \circ \pi_{pp'} =: \pi_{pq}$

$$|\pi_{pq}| = |\pi_{pp'}| + |p'q'| + |\pi_{q'q}| \leq t\,|pp'| + |p'q'| + t\,|q'q|$$

$$\leq t\,\frac{2}{s}|pq| + \left(1+\frac{4}{s}\right)|pq| + t\,\frac{2}{s}|pq|$$

$$= \left(\frac{4(t+1)}{s} + 1\right)|pq| \overset{!}{\underset{\text{Def } t}{=}} t\,|pq|. \qquad \square$$

$\frac{s+4}{s-4} = t = 1+\varepsilon \Rightarrow \cancel{s}+4 = \cancel{s}+s\varepsilon-4-4\varepsilon \Rightarrow (s-4)\varepsilon = 8 \Rightarrow \varepsilon = \frac{8}{s-4}$

$s = \frac{8}{\varepsilon}+4$

$\Rightarrow \frac{8^d}{\varepsilon^d} \leq s^d \leq \left(2\cdot\frac{8}{\varepsilon}\right)^d = \frac{16^d}{\varepsilon^d}$

$\frac{4(t+1)}{s} + 1 = \frac{s+4}{s-4} = t$, denn $t+1 = \frac{s+4}{s-4} + 1 = \frac{s+4+s-4}{s-4} = \frac{2s}{s-4}$

14.7.1

$$\frac{4(t+1)}{s} + 1 \quad \stackrel{!}{=} \quad t \quad = \frac{s+4}{s-4}$$

$$t+1 = \frac{2s}{s-4}$$

$$\frac{4(t+1)}{s} = \frac{8}{s-4}$$

$$\frac{4(t+1)}{s} + 1 = \frac{s+4}{s-4}$$

more applications

→ • closest pair

→ • k closest pairs

→ • all nearest neighbors

→ • approximation of MST (choose MST of a spanner)

→ • approximately computing the dilation of N

• adding an edge that maximally reduces the dilation of N (approx.)

→ p 18.1

• given $a \neq b \in S$, find the unique i s.th. $a \in A_i$, $b \in B_i$: $O(\log n)$ after preprocessing Split Tree in $O(s^d n \log n)$ or $O(n)$

Corollary 1 Constructing WSPD is in $\Theta(n \log n)$

Proof Closest pair (in $d=1$) is in $\Omega(n \log n)$, by reduction from ε-closeness. □

[Theorem 3 can be generalized to

- all nearest neighbors
(to each $p \in S$, find nearest neighbor in S)
- k closest pairs
$(\underbrace{|p_1 q_1| \le |p_2 q_2| \le \ldots \le |p_k q_k|}_{\text{report}} \le |p_{k+1} q_{k+1}| \ldots)$]

applications of spanners

besides the natural one

Euclidean minimum spanning tree of n points in $\mathbb{R}^d$

running Kruskal or other graph algorithm on complete graph:
$\Omega(\#\text{edges}) = \Omega(n^2)$. (better: $\Omega(d n^2)$)

$d=2$: MST edges $\subseteq$ Delaunay triangulation

in general: MST in $O(|E| \log E)$

only $O(n)$ edges

$\rightsquigarrow$ $O(n \log n)$ algorithm

(also closest pair, all nearest neighbors can be nicely solved by Voronoi diagram / Delaunay triangulation)

$d>2$ worst case complexity of Voronoi diagram grows with $n^{\lfloor \frac{d}{2} \rfloor}$ $\rightsquigarrow$ no use.

spanners can help!

jetzt wieder d, ε fest

Theorem 4 $S \subseteq \mathbb{R}^d$, $|S| = n$, $\varepsilon > 0$ fixed.

Can compute in time $O(n \log n)$ spanning tree T of S such that weight$(T) \leq (1+\varepsilon)$ weight$(MST(S))$.

Proof

- compute spanner N of S with dilation $1+\varepsilon$, $O(n)$ edges — $O(n \log n)$
- compute minimum spanning tree T of network N — $O(n \log n)$

Let M the real MST of S

for each edge $e_i = (p_i, q_i)$ of M

there is path π_i from p_i to q_i in N

satisfying $|\pi_i| \leq (1+\varepsilon)|p_i q_i|$

N is the set of min-edges of an $s>2$ WSPD

$\Downarrow$

$T = $ Euclidean MST ???

not clear how to produce quickly! $\Rightarrow$ MST M can be completed into $1+\varepsilon$-spanner

$G :=$ union of all paths π_i

$\Rightarrow$ G connected graph over vertex set S in N

$$\Rightarrow |T| \leq |G| \leq \sum_{i=1}^{n-1} |\pi_i| \leq (1+\varepsilon) \sum_{i=1}^{n-1} |p_i q_i|$$

T is MST of N, $G \subseteq N$ and spans

$$= (1+\varepsilon)|M|$$

$\boxed{\text{Th 4}}$

Better still:

One can find spanners of dilation $1+\varepsilon$, $O(n)$ edges, weight $\leq$ constant $\cdot$ weight(MST), degree $\leq$ constant in time $O(n \log n)$!

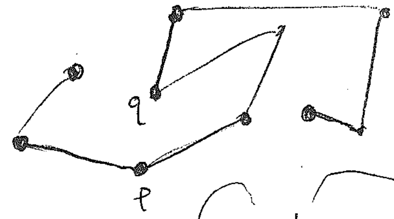
Aber i.a. keine Bäume

(Das, Heffernan, Narasimhan, Salowe '93, '95)

another spanner application

approximating the dilation of a given graph.

here: graph = simple polygonal chain C over n vertices



railway

naive: dilation $\delta(C)$ in time $O(n^2)$

Hallen $O(n \log n)$ randomisiert gesehen

Faster approximation algorithm:

Given chain (=path) C over vertices $V = \{p_0, \dots, p_{n-1}\}$

- compute $1+\varepsilon$-spanner G of V: Theorem 2. with $O(n)$ many edges
- for each edge $e=(p,q)$ of G: compute $\delta_C(p,q)$ — $O(n)$ total time, because $O(n)$ edges, each needing $O(1)$ time (not true for general graphs)
- output $\tilde{\delta}(C)$ = maximum of these values

!

$$\Rightarrow \quad \delta(C) \le (1+\varepsilon)\,\tilde{\delta}(C) \underset{\text{trivial}}{\le} (1+\varepsilon)\,\delta(C)$$

$1+\varepsilon$-approx

Proof: Assume $\delta(C) = \delta_C(p,q)$, p,q vertices of C

$\Rightarrow$ for shortest p-to-q path $\pi_{pq} = (e_1, e_2, \dots, e_r)$ in spanner G: $\dfrac{|\pi_{pq}|}{|pq|} \le 1+\varepsilon$

let $e_i = (q_i, q_{i+1})$ and $C_{q_i}^{q_{i+1}}$ the piece of C connecting q_i, q_{i+1}

$$\Rightarrow \quad \delta(C) = \delta_C(p,q) = \frac{|C_p^q|}{|pq|} \le \frac{\sum_{i=1}^{r} |C_{q_i}^{q_{i+1}}|}{\frac{\sum_{i=1}^{r} |e_i|}{1+\varepsilon}}$$

$$= (1+\varepsilon) \frac{\sum_i |C_{q_i}^{q_{i+1}}|}{\sum_i |e_i|} \le (1+\varepsilon) \max_i \frac{|C_{q_i}^{q_{i+1}}|}{|e_i|}$$

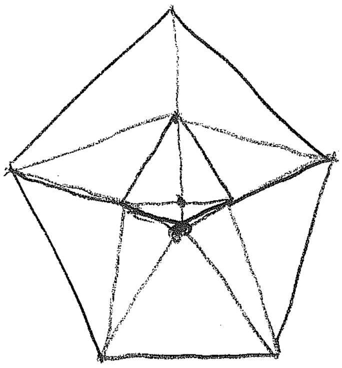
$$= (1+\varepsilon) \max_i \delta_C(q_i, q_{i+1}) \le (1+\varepsilon)\,\tilde{\delta}(C) \quad \square$$

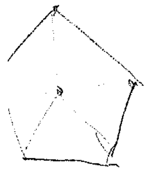
gives $(1+\varepsilon)$ approximation in time $O(\frac{1}{\varepsilon^2} n \log n)$

später: korrekter Wert in $O(n \log n)$

Known

1





.05146

1.02046

D. Lorenz

Classic

Given: finite point set $S \subset \mathbb{R}^2$

Find planar graph (S,E) of smallest possible dilation

($\rightsquigarrow$ Jit: sup dilation value

MinDilTriangulation: status?

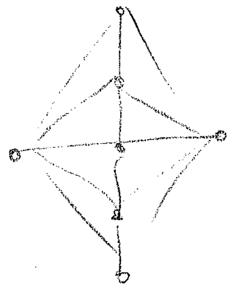
but can test all triang.)

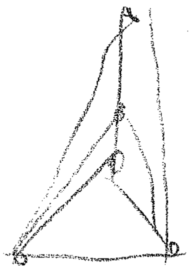
Modif. (V,E)

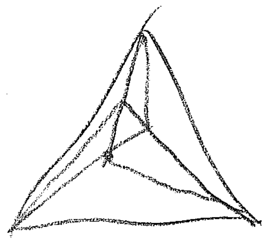
where $S \subset V$

2 (M. Kutz, RK)

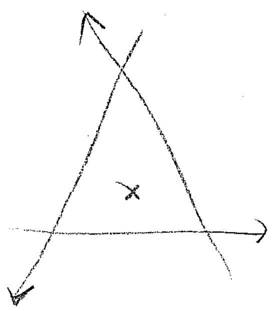
S not special

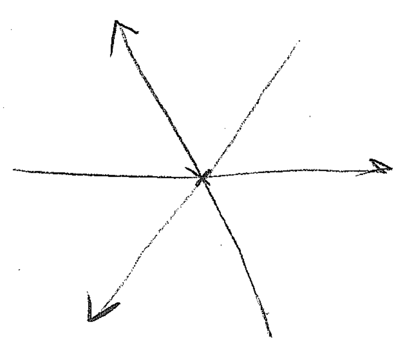


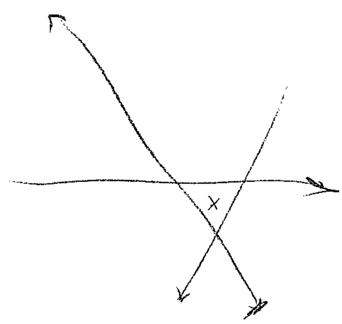


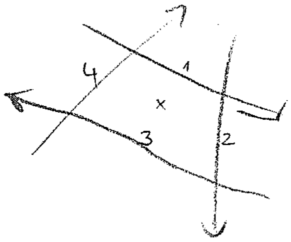


$\Rightarrow \exists\, \delta(S) > 1 \quad \forall\, \underset{\text{planar}}{(V,E)},\ V \supseteq S: \quad \delta((V,E)) \geq \delta(S)$

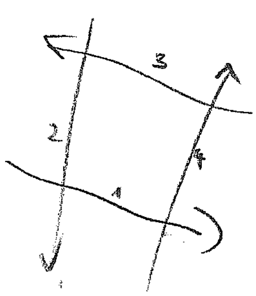








$(1,2), (2,3), (3,4), (4,1)$

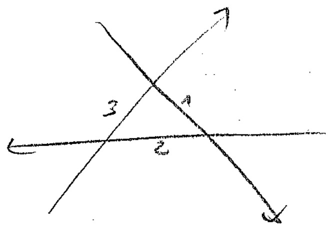


$(1,4), (4,3), (3,2), (2,1)$

edge $e = (C_i, C_j), (C_j, C_l)$ feasible at time t

$\Leftrightarrow$ ∃ robot path ~~...~~
~~... at $C_i \cap C_j$~~,
starting at time t at $C_i \cap C_j$
using C_j
arriving at $C_j \cap C_l$ at $t+\delta$

Then $\delta = w_t(e)$



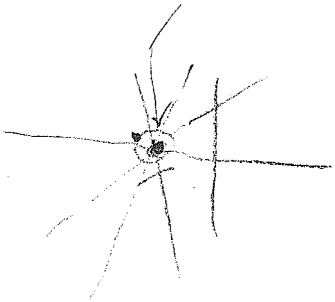
$(1,2)$ $(2,3)$ $(3,1)$

feasible(e) = die t, bei denen man in endlicher Zeit so von $C_i \cap C_j$ nach $C_j \cap C_l$ kommt

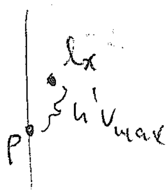
$h = \frac{D}{4 v_{max}}$

D = min Abstand Zero Punkt zu Carriers $\not\ni$ Zero Punkt at Zero Zeit

Zero Zeit



$t_x - h'$



at t_x: $|p - l_x| < 2h' v_{max}$
$< 2h v_{max} = \frac{D}{2}$

Zero Zeit