How to Grow Your Balls

(Distance Oracles Beyond the Thorup–Zwick Bound)

Mihai Pătrașcu
at&t

Liam Roditty
Bar-Ilan University

FOCS'10

Distance Oracles

Distance oracle = replacement of APSP matrix

[Thorup, Zwick STOC'01] For k=1,2,3,...:

Preprocess undirected, weighted graph G to answer:

query(s,t): return đ with $d_G(s,t) \leq$ đ $\leq (2k\text{-}1) \cdot d_G(s,t)$

Space $O(n^{1+1/k})$ with O(1) query time

Approximation	Space
1	n^2
3	$n^{1.5}$
5	$n^{4/3}$
$2k$-1	$n^{1+1/k}$

Two Sides of Distance Oracles

Compression question:

Encode dense graphs with $\tilde{O}(n^{1+1/k})$ bits,
such that $(2k\text{-}1)$-apx distances can be retrieved

[Matoušek'96] (∀) finite metric space on n points
$\mapsto \ell_\infty$ with dimension $O(k \cdot n^{1+1/k} \lg n)$ with distortion $2k\text{-}1$

Spanners: (∀) unweighted graph G
(∃) $H \subseteq G$ with $O(n^{1+1/k})$ edges: $d_G \le d_H \le (2k\text{-}1)\, d_G$

Data structures question:

A data structure of size $\tilde{O}(n^{1+1/k})$
can answer $(2k\text{-}1)$-apx distance queries in constant time

Two Sides of Distance Oracles

Compression question:

Encode dense graphs with $\tilde{O}(n^{1+1/k})$ bits,

such that $(2k\text{-}1)$-apx distances can be retrieved

[Matoušek'96] (∀) finite metric space on n points

$\mapsto \ell_\infty$ with dimension $O(k \cdot n^{1+1/k} \lg n)$ with distortion $2k\text{-}1$

Spann

(

Optimal space, assuming:

Girth conjecture: [Erdős] et al.

(∃) graphs with $\Omega(n^{1+1/k})$ edges and girth $2k+2$

Data

A

can answer $(2k\text{-}1)$-apx distance queries in constant time

Two Sides of Distance Oracles

Comp

E

[Mat

Span

[Sommer, Verbin, Yu FOCS'09]

Assume $m=\tilde{O}(n)$ – sparse graphs.

If c-apx distance queries take O(1) time

$\Rightarrow$ the space is $\geq n^{1+\Omega(1/c)}$

Beat [Thorup, Zwick] for graphs with $\ll n^{1+1/k}$ edges?

$(\exists)$ H $\subseteq$ G with $O(n^{1+1/k})$ edges: $d_G \leq d_H \leq (2k\text{-}1)\, d_G$

Data structures question:

A data structure of size $\tilde{O}(n^{1+1/k})$

can answer $(2k\text{-}1)$-apx distance queries in constant time

Two Sides of Distance Oracles

Co

Many other spanners:

[BKMP'05] $d_H \le d_G + 6$ with $O(n^{4/3})$ edges

[EP'01] $d_H \le (1+\varepsilon)\, d_G + O(1)$ with $O(n^{1+\delta})$ edges

…

Use additive approximation in distance oracles?

[M

Spanners: (∀) unweighted graph G

(∃) $H \subseteq G$ with $O(n^{1+1/k})$ edges: $d_G \le d_H \le (2k\text{-}1)\, d_G$

Data structures question:

A data structure of size $\tilde{O}(n^{1+1/k})$

can answer $(2k\text{-}1)$-apx distance queries in constant time

New Upper Bounds

Unweighted graphs:

Preprocess any graph G

⇒ distance oracle of size $O(n^{5/3})$ that finds distance $\leq 2d_G + 1$

Weighted graphs:

Preprocess G with $m=n^2/\alpha$ edges

⇒ distance oracle of size $O(n^2 / \alpha^{1/3})$ with approximation 2

Can report path in O(1) / edge.

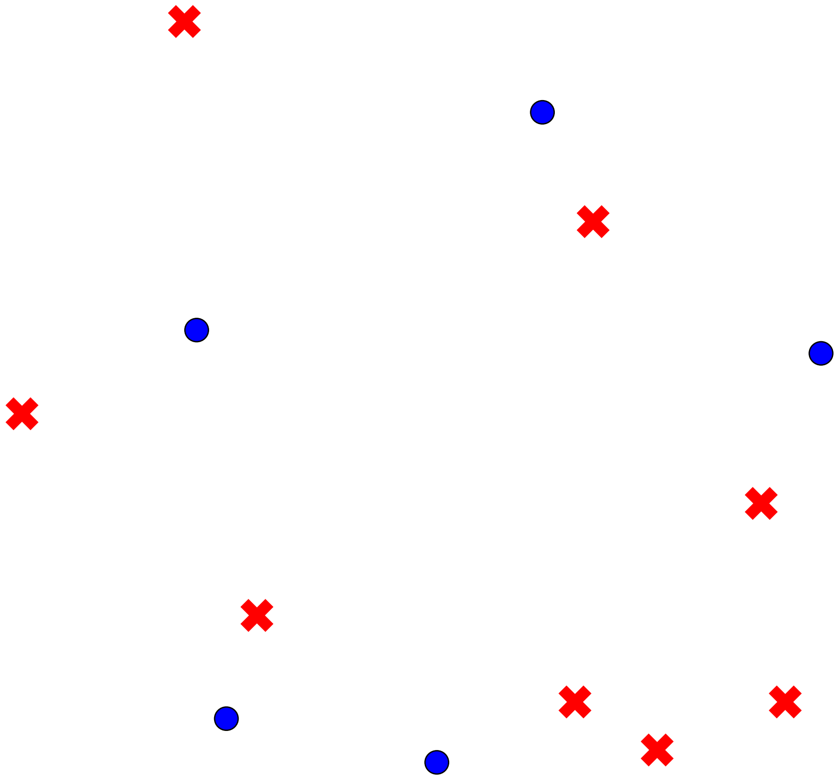
The Algorithm

A = {✖} = sample $n^{2/3}$ vertices

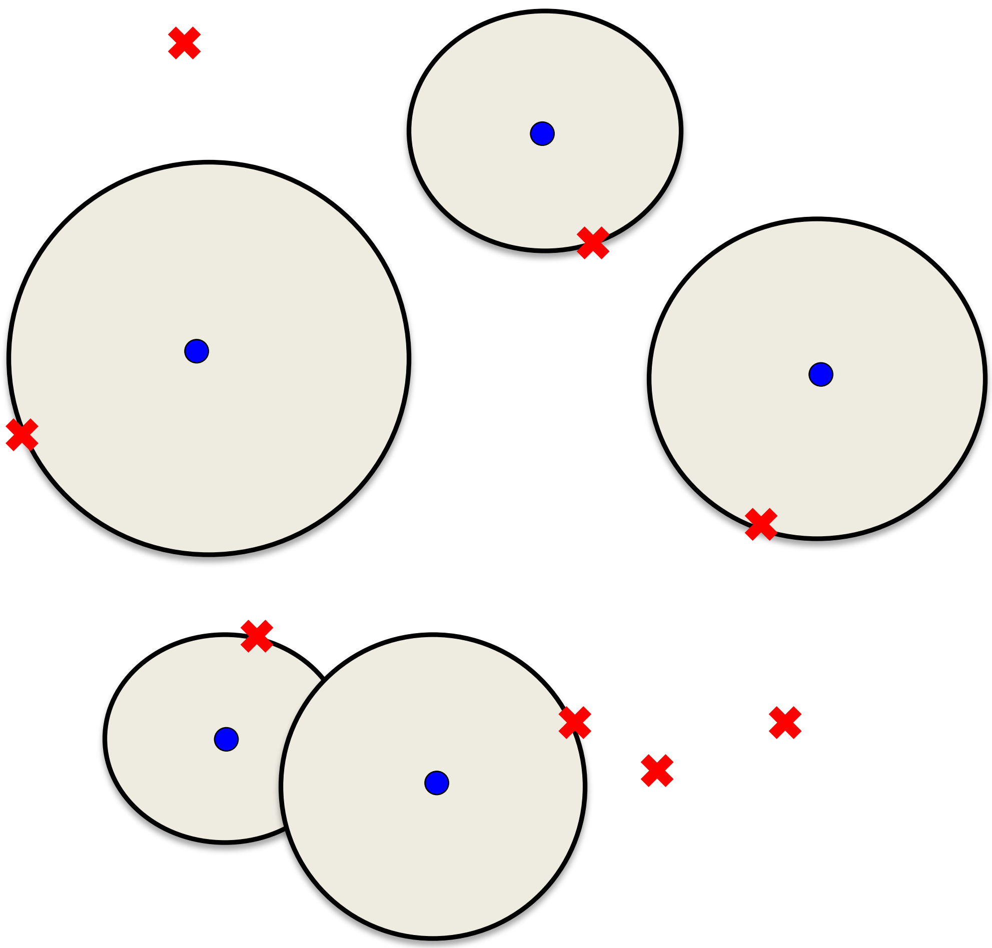
The Algorithm

A = {✖} = sample $n^{2/3}$ vertices

B = {●} = sample $n^{1/3}$ vertices



The Algorithm



A = {✖} = sample $n^{2/3}$ vertices

B = {●} = sample $n^{1/3}$ vertices

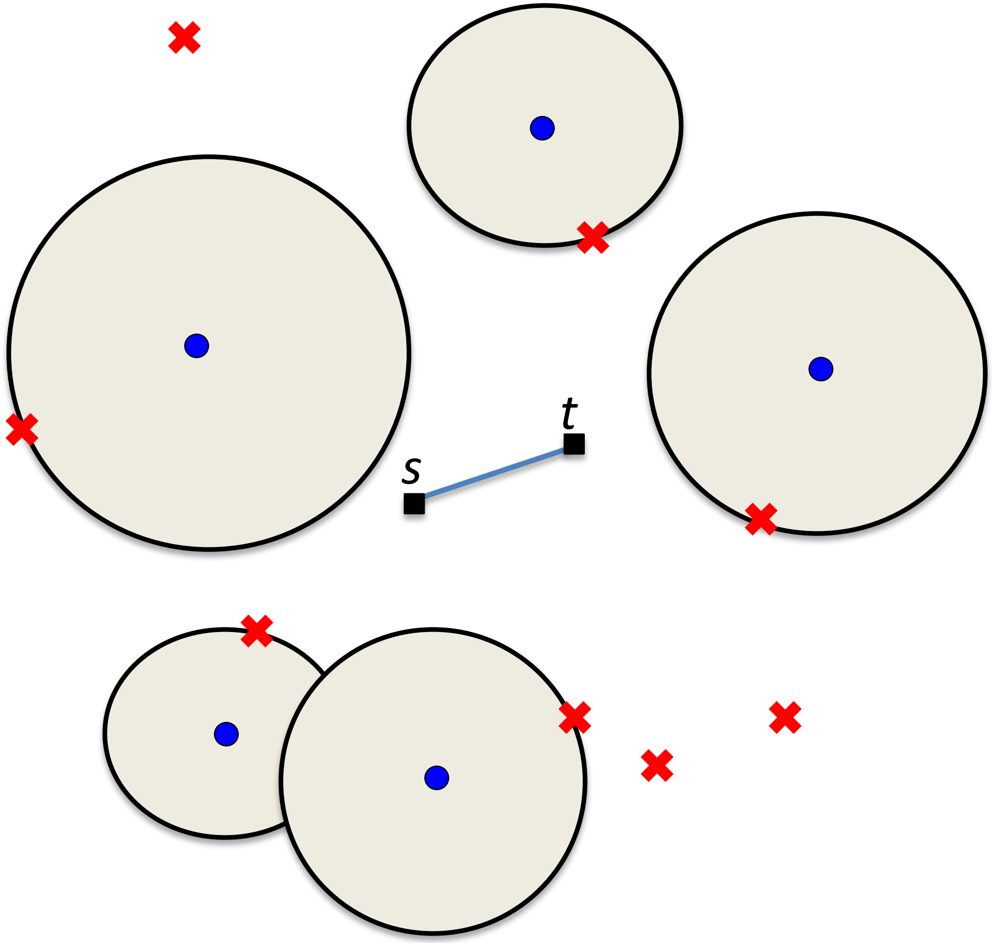
C = $\cup_{u \in B}$ Ball(u → A)

$E[|C|] = n^{2/3}$

Data structure:

- distances C ↔ V

The Algorithm



A = {✖} = sample $n^{2/3}$ vertices

B = {●} = sample $n^{1/3}$ vertices

C = $\cup_{u \in B}$ Ball($u \rightarrow$ A)

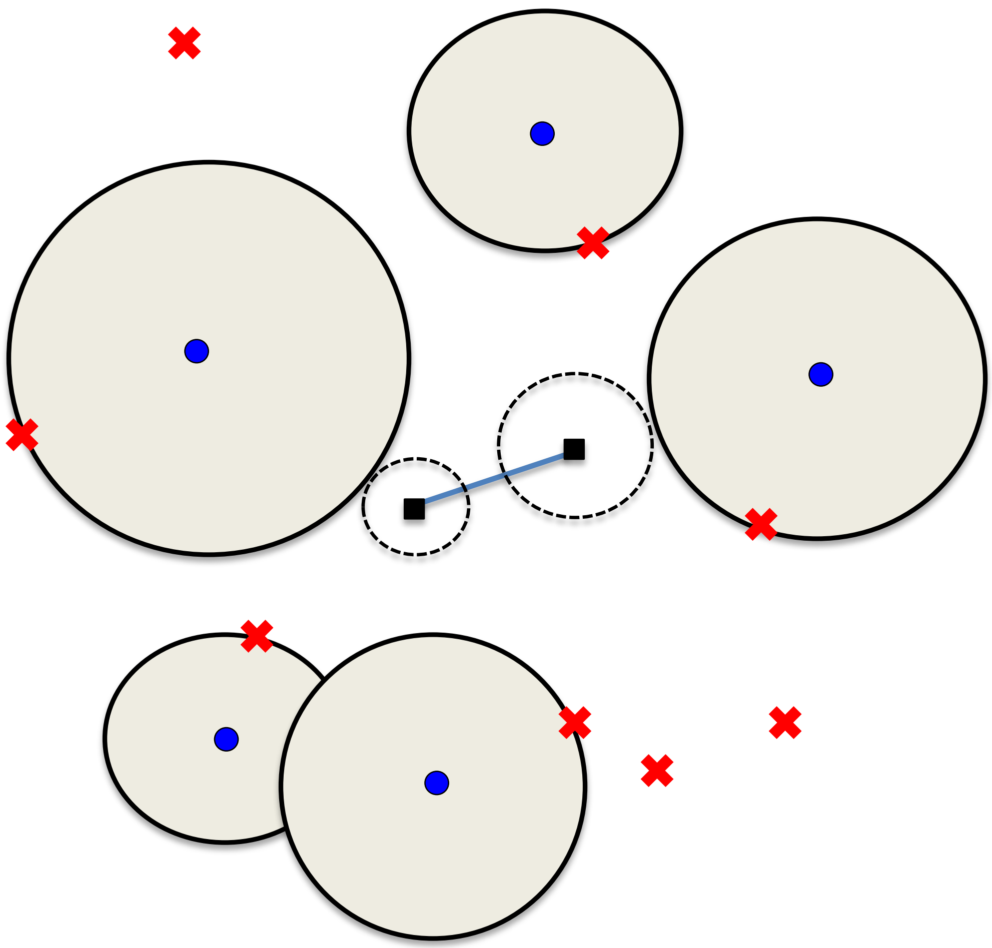
E[|C|]=$n^{2/3}$

Data structure:

- distances C ↔ V

Query(s,t):

The Algorithm



A = {✖} = sample $n^{2/3}$ vertices

B = {●} = sample $n^{1/3}$ vertices

C = $\bigcup_{u \in B}$ Ball($u \rightarrow$ A)

E[|C|]=$n^{2/3}$

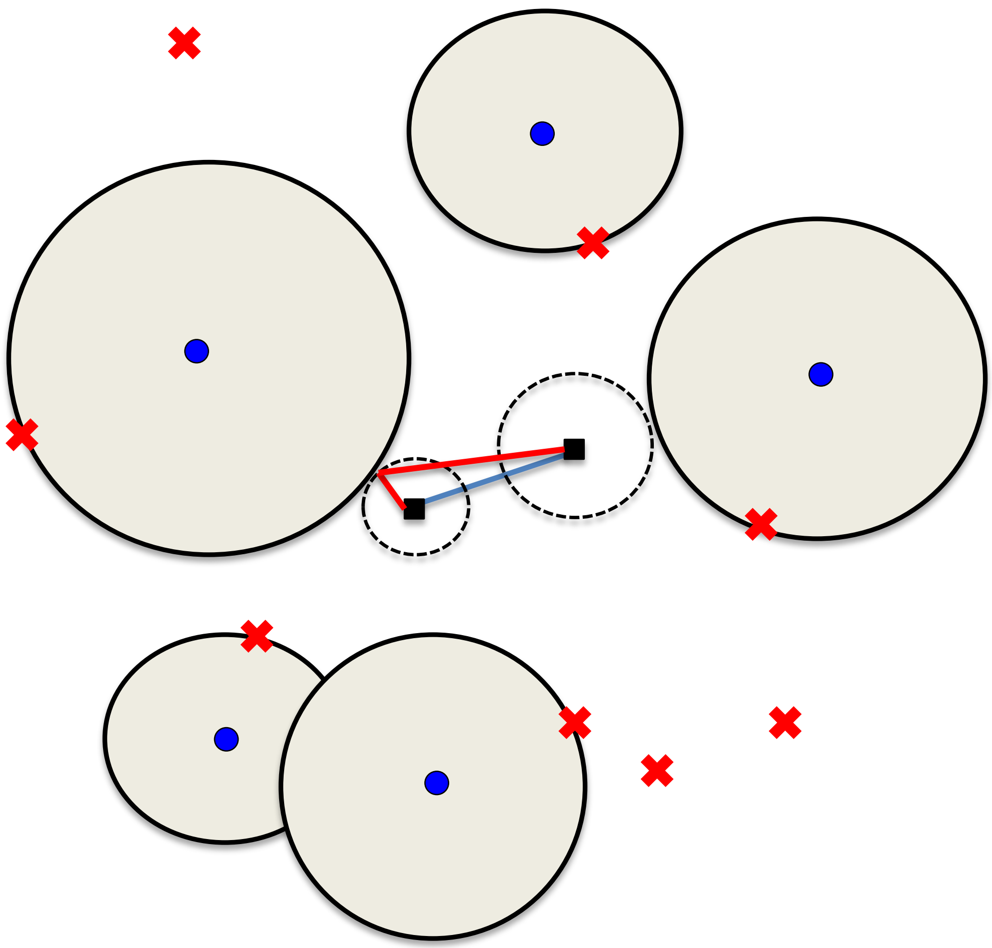
Data structure:

- distances C ↔ V

Query(s,t):

- R_s, R_t = distance to NN in C
- If Ball(s, R_s) ∩ Ball(t, R_t) = ∅ ⇒ min { R_s, R_t } ≤ ½ d(s,t)

The Algorithm



A = {✖} = sample $n^{2/3}$ vertices

B = {●} = sample $n^{1/3}$ vertices

C = $\bigcup_{u \in B}$ Ball(u → A)

E[|C|]=$n^{2/3}$

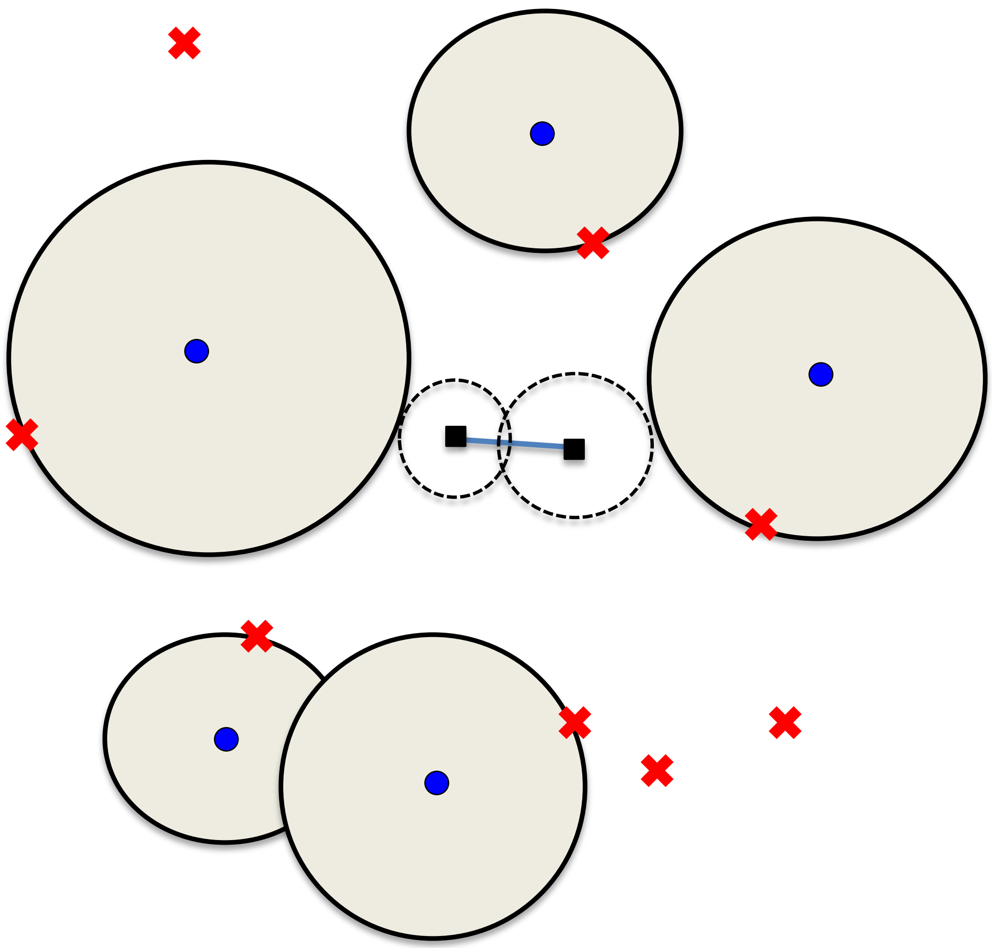
Data structure:

- distances C ↔ V

Query(s,t):

- R_s, R_t = distance to NN in C
- If Ball(s, R_s) ∩ Ball(t, R_t) = ∅
  ⇒ min { R_s, R_t } ≤ ½ d(s,t) ✔

The Algorithm



A = {✖} = sample $n^{2/3}$ vertices

B = {●} = sample $n^{1/3}$ vertices

C = $\bigcup_{u \in B}$ Ball(u → A)

E[|C|]=$n^{2/3}$

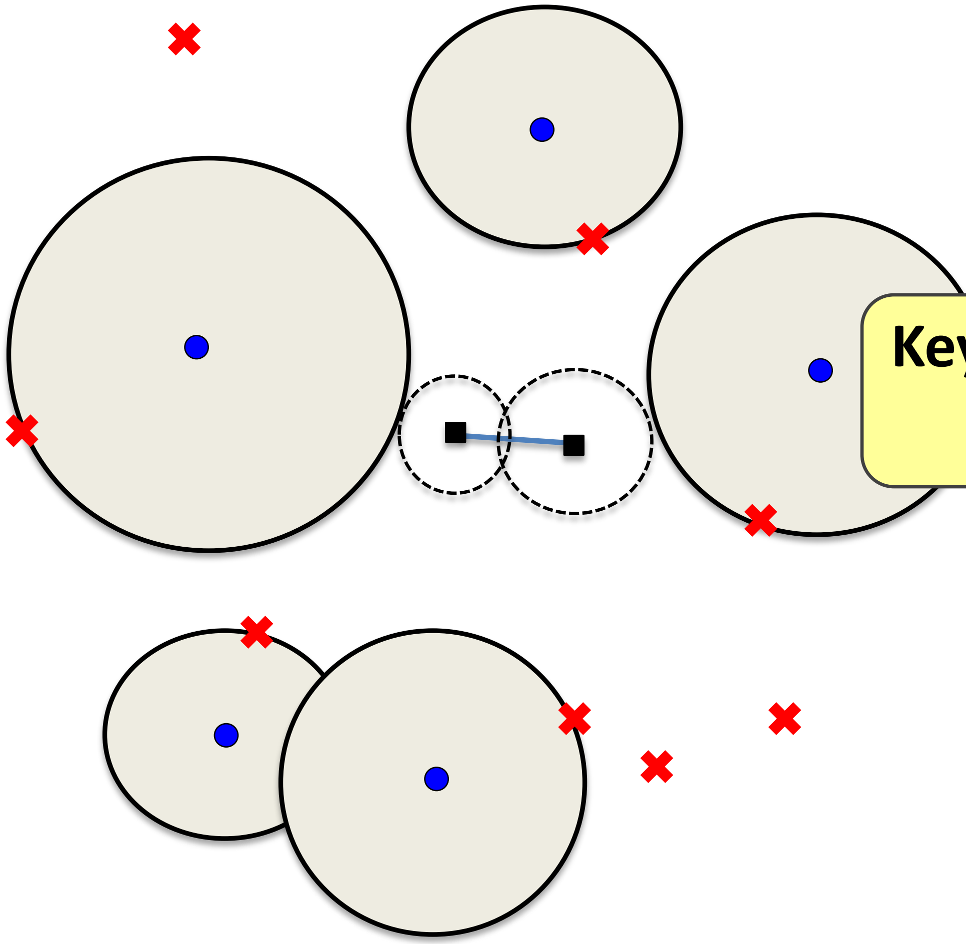
Data structure:

- distances C ↔ V
- pairs when Ball(s)∩Ball(t)

Query(s,t):

- R_s, R_t = distance to NN in C
- If Ball(s, R_s) ∩ Ball(t, R_t) = ∅
  ⇒ min { R_s, R_t } ≤ ½ d(s,t) ✔
- If Ball(s, R_s) ∩ Ball(t, R_t) ≠ ∅

The Algorithm



A = {✖} = sample $n^{2/3}$ vertices

B = {●} = sample $n^{1/3}$ vertices

C = $\bigcup_{u \in B}$ Ball($u \rightarrow$ A)

E[|C|]=$n^{2/3}$

Key observation:

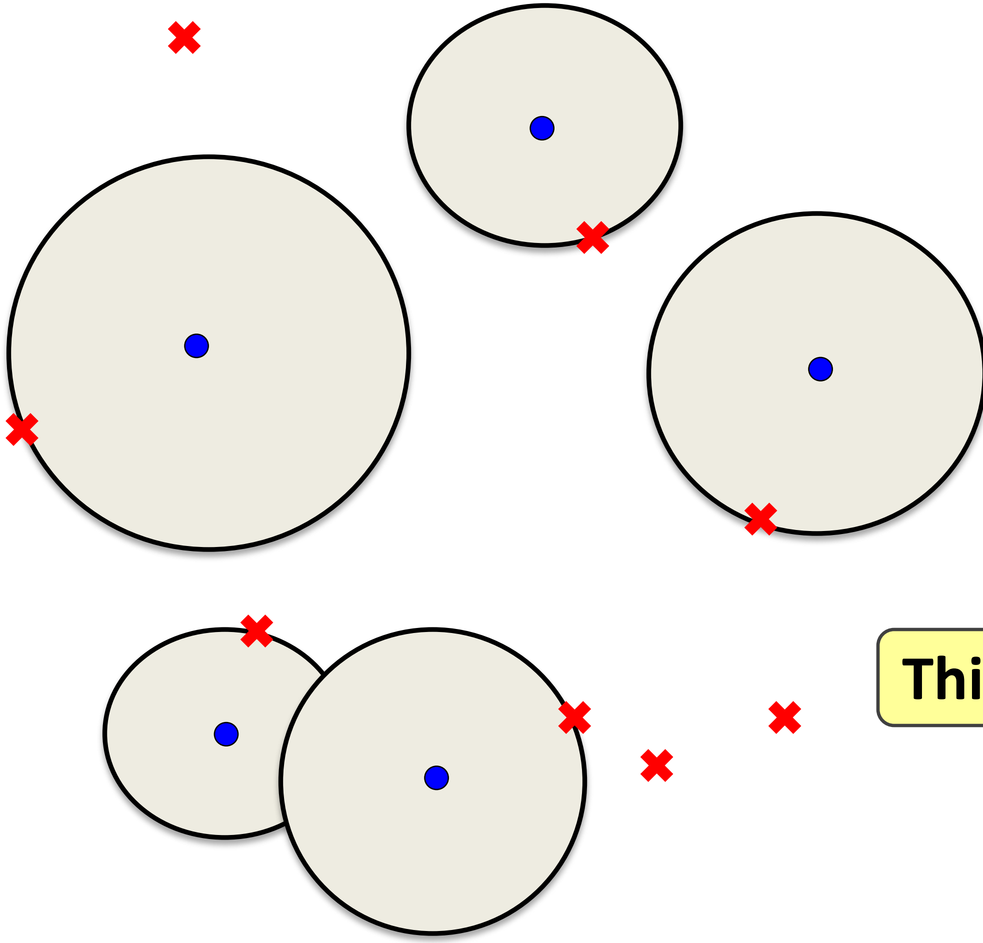
E[#pairs] = O($n^{5/3}$)

- pairs when Ball(s)∩Ball(t)

Query(s,t):

- R_s, R_t = distance to NN in C
- If Ball(s, R_s) ∩ Ball(t, R_t) = Ø
  ⇒ min { R_s, R_t } ≤ ½ d(s,t) ✔
- If Ball(s, R_s) ∩ Ball(t, R_t) ≠ Ø

The Algorithm



A = {✖} = sample $n^{2/3}$ vertices

B = {●} = sample $n^{1/3}$ vertices

C = $\bigcup_{u \in B}$ Ball($u \to$ A)

$E[|C|]=n^{2/3}$

Data structure:

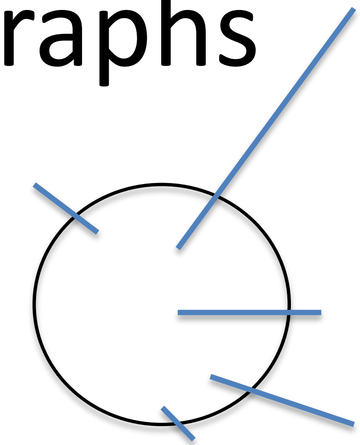
- distances C ↔ V
- pairs when Ball(s)∩Ball(t)

Query(s,t):

This is a lie.

- R_s, R_t = distance to NN in C
- If Ball(s, R_s) ∩ Ball(t, R_t) = Ø ⇒ min { R_s, R_t } ≤ ½ d(s,t) ✔
- If Ball(s, R_s) ∩ Ball(t, R_t) ≠ Ø ✔

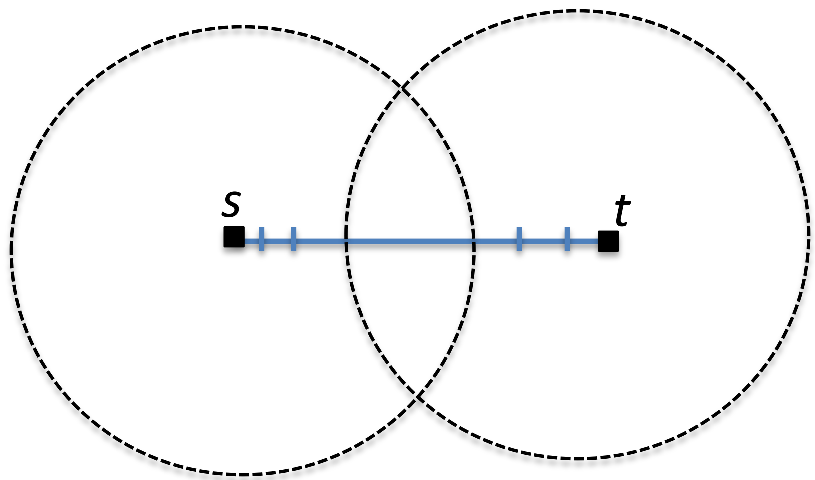
Geometric balls ≠ Balls in graphs



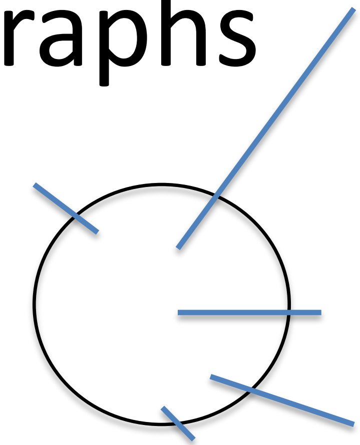
Weighted graphs:

Ball(s,r)={edges adjacent to vertices at distance ≤ *r*}

To bound ball, sample *edges* ⇒ sparsity matters!



Geometric balls ≠ Balls in graphs



Weighted graphs:

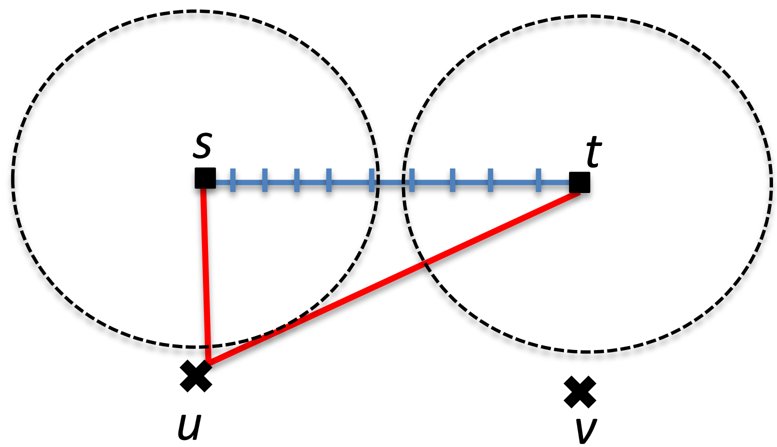
Ball(s,r)={edges adjacent to vertices at distance ≤ *r*}

To bound ball, sample *edges* ⇒ sparsity matters!

Unweighted graphs:

d(*s*,*u*) = d(*t*,*v*) = ⌈ ½ d(*s*,*t*) ⌉

Just accept additive 1…



Upper Bounds

Can we get rid of "+1" for not-too-dense graphs?

Unweighted graphs:

Preprocess any graph *G*

⇒ distance oracle of size $O(n^{5/3})$ that finds distance ≤ $2d_G + 1$

Better bounds?

Weighted graphs:

Preprocess *G* with $m=n^2/\alpha$ edges

⇒ distance oracle of size $O(n^2 / \alpha^{1/3})$ with approximation 2

Milder dependence on *m*?

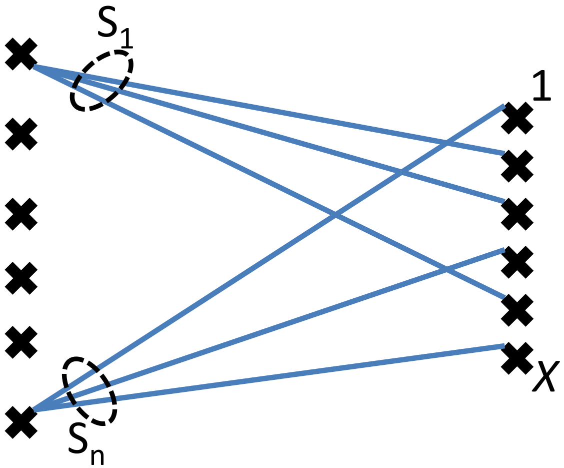
E.g. $O(m + n^{5/3})$

Set-Intersection Hardness

Preprocess $S_1, \ldots, S_n \subseteq [X]$
query(i,j): is $S_i \cap S_j = \emptyset$?

Conjecture: Let $X = \lg^{O(1)} n$. If query time = O(1), space = $\tilde{\Omega}(n^2)$

Even in sparse graphs, approximation **< 2** requires $\tilde{\Omega}(n^2)$ space



Set-Intersection Hardness

Preprocess $S_1, \ldots, S_n \subseteq [X]$

query(i,j): is $S_i \cap S_j = \emptyset$?

Conjecture: Let $X = \lg^{O(1)} n$. If query time = O(1), space = $\tilde{\Omega}(n^2)$

↓

Even in sparse graphs, approximation **< 2** requires $\tilde{\Omega}(n^2)$ space

↓

Apx. $(2\text{-}\varepsilon)d_G + O(1)$ in unweighted graphs requires $\tilde{\Omega}(n^2)$ space

Set-Intersection Hardness

Preprocess $S_1, \ldots, S_n \subseteq [X]$
query(i,j): is $S_i \cap S_j = \emptyset$?

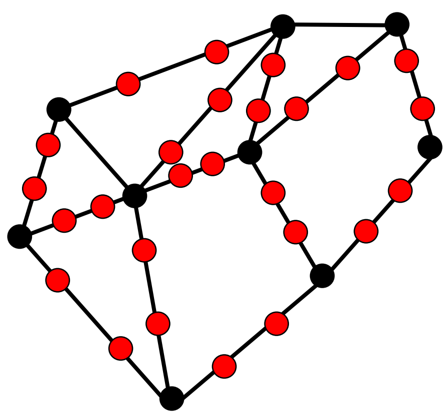
Conjecture: Let $X = \lg^{O(1)} n$. If query time = O(1), space = $\tilde{\Omega}(n^2)$

↓

Even in sparse graphs, approximation **< 2** requires $\tilde{\Omega}(n^2)$ space

↓

Apx. $(2\text{-}\varepsilon)d_G + O(1)$ in unweighted graphs requires $\tilde{\Omega}(n^2)$ space



New Lower Bounds

Preprocess $S_1, \ldots, S_n \subseteq [X]$
query(i,j): is $S_i \cap S_j = \emptyset$?

Conjecture: Let $X = \lg^{O(1)} n$. If query time = O(1), space = $\tilde{\Omega}(n^2)$

Actually, randomized conjecture…

↓

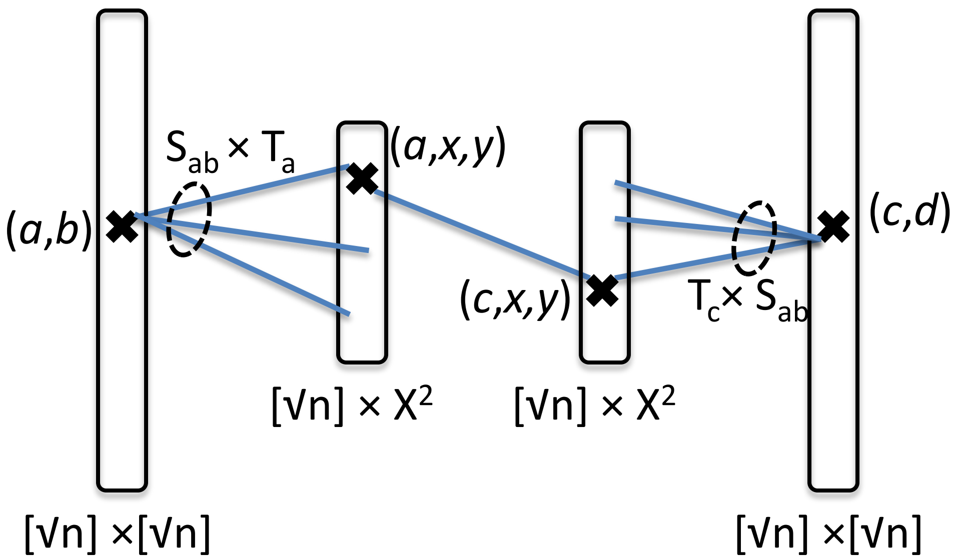
In unweighted graphs with $m=n^2/\alpha$ edges,
constant-time 2-approximation needs space $\tilde{\Omega}(n^2 / \sqrt{\alpha})$

↓

Constant-time approximation $2d_G$+O(1) needs space $\tilde{\Omega}(n^{1.5})$

Lower Bound

In graphs with $\tilde{O}(n)$ edges, 2-apx needs space $\tilde{\Omega}(n^{1.5})$

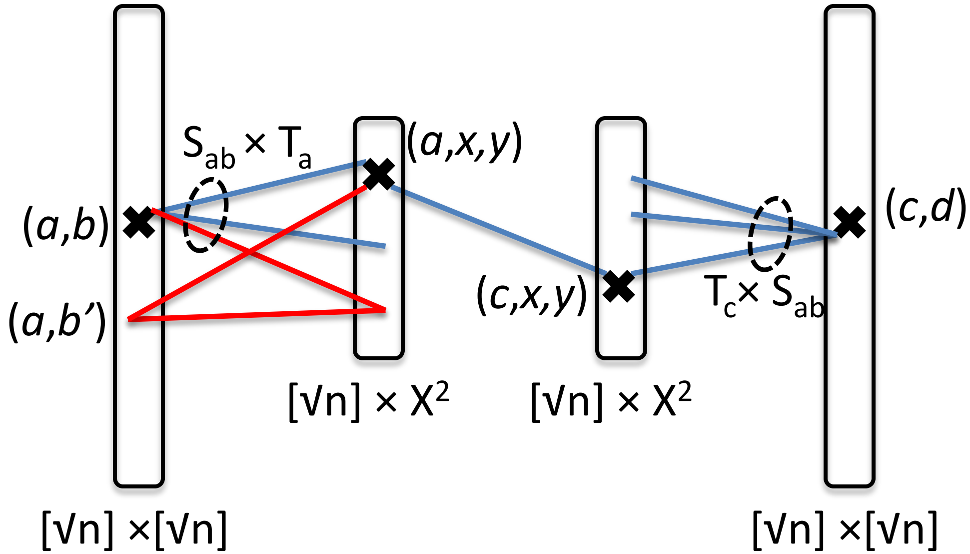


$(S_{ab} \cap T_c \neq \emptyset) \wedge (T_a \cap S_{cd} \neq \emptyset)$
$\Rightarrow$ distance = 3

3-apx $\Rightarrow$ distance ≤ 5

Lower Bound

In graphs with $\tilde{O}(n)$ edges, 2-apx needs space $\tilde{\Omega}(n^{1.5})$



$(S_{ab} \cap T_c \neq \emptyset) \wedge (T_a \cap S_{cd} \neq \emptyset)$
$\Rightarrow$ distance = 3

3-apx $\Rightarrow$ distance ≤ 5

$\Rightarrow$ either $T_a \cap S_{cd} \neq \emptyset$
or $S_{ab} \cap T_c \neq \emptyset$

The End

Recent progress:

- space $\Omega(n^{5/3})$ for 2-apx
- many result for apx > 2

My flight is at 4:36.

Questions @ 857-253-1282