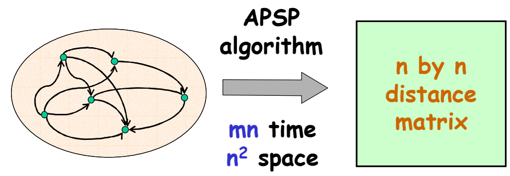
Approximate All-Pairs shortest paths
Approximate distance oracles
Spanners and Emulators

Uri Zwick
Tel Aviv University

Summer School on Shortest Paths (PATH05)
DIKU, University of Copenhagen

All-Pairs Shortest Paths



Input: A weighted directed graph $G=(V,E)$, where $|E|=m$ and $|V|=n$.

Output: An $n \times n$ distance matrix.

Approximate Shortest Paths

Let $\delta(u,v)$ be the distance from u to v.

Multiplicative error

An estimated distance $\delta'(u,v)$ is of **stretch** t iff

$$\delta(u,v) \le \delta'(u,v) \le t \cdot \delta(u,v)$$

Additive error

An estimated distance $\delta'(u,v)$ is of **surplus** t iff

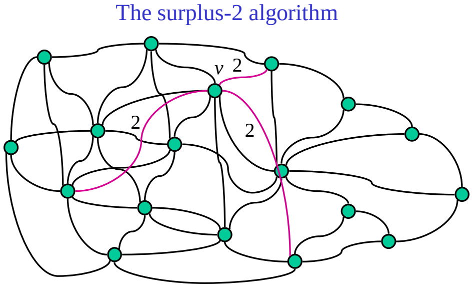
$$\delta(u,v) \le \delta'(u,v) \le \delta(u,v) + t$$

All-Pairs Almost Shortest Paths

unweighted, undirected graphs

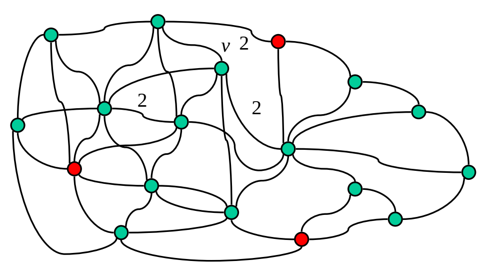
Surplus	Time	Authors
0	mn	folklore
2	$n^{5/2}$	Aingworth-Chekuri-Indyk-Motwani '96
2	$n^{3/2}m^{1/2}$	Dor-Halperin-Zwick '96
2	$n^{7/3}$	"
$2(k-1)$	$n^{2-1/k}m^{1/k}$	"
$2(k-1)$	$n^{2+1/(3k-4)}$	"

The surplus-2 algorithm



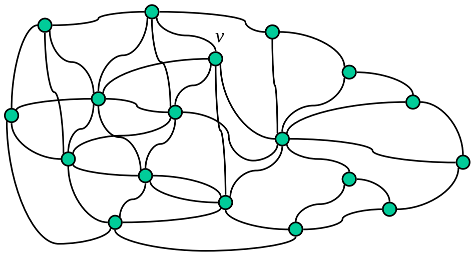
- For each non-selected vertex v, run an SSSP algorithm on the subgraph composed of the selected edges, augmented by new edges connecting v to all the selected vertices.
- Select each vertex independently with probability $p=(n/m)^{1/2}$
- For each vertex adjacent to a selected vertex, select an edge connecting it to a selected vertex.
- Run a SSSP algorithm from each selected vertex.
- For each vertex non-adjacent to a selected vertex, select **all** its edges.
- The length of each new edge is the distance between the endpoints in the original graph.

The surplus-2 algorithm



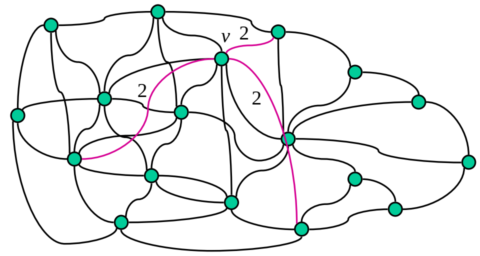
- Select each vertex independently with probability $p=(n/m)^{1/2}$
- Run a SSSP algorithm from each selected vertex.

The surplus-2 algorithm



- For each vertex adjacent to a selected vertex, select an edge connecting it to a selected vertex.
- For each vertex non-adjacent to a selected vertex, select **all** its edges.

The surplus-2 algorithm

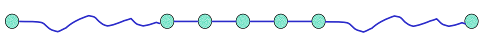


- For each non-selected vertex *v* , run an SSSP algorithm on the subgraph composed of the selected edges, augmented by new edges connecting *v* to all the selected vertices.
  - The length of each new edge is the distance between the endpoints in the original graph.

The Surplus-2 algorithm

Correctness – Case 1

Case 1: All vertices on a shortest path from *u* to *v* do not have selected neighbors.



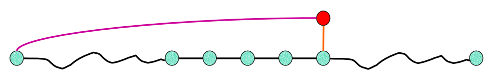
All the edges on the path are selected.

We find a shortest path from *u* to *v*.

The Surplus-2 algorithm

Correctness – Case 2

Case 2: At least one vertex on a shortest path from *u* to *v* has a selected neighbor.

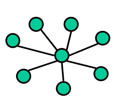


Consider the last vertex with a selected neighbor.

We find a path from *u* to *v* of surplus at most 2

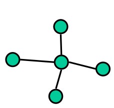
The Surplus-2 algorithm - Analysis

HIGH degree vertex



With high probability at least one neighbor is selected

low degree vertex



Taking all adjacent edges is relatively cheap

The expected number of the edges selected is $O((mn)^{1/2})$

The expected running time is therefore $O(n(mn)^{1/2})=O(n^{3/2}m^{1/2})$

All-Pairs Almost Shortest Paths

unweighted, undirected graphs

Surplus	Time	Authors
0	*mn*	folklore
2	$n^{5/2}$	Aingworth-Chekuri-Indyk-Motwani '96
2	$n^{3/2}m^{1/2}$	Dor-Halperin-Zwick '96
2	$n^{7/3}$	”
2(k-1)	$n^{2-1/k}m^{1/k}$	”
2(k-1)	$n^{2+1/(3k-4)}$	”

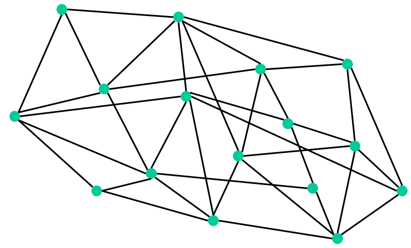
All-Pairs Almost Shortest Paths

weighted undirected graphs

Stretch	Time	Reference
1	mn	Dijkstra + FT
2	$n^{3/2}m^{1/2}$	Cohen-Zwick '97
7/3	$n^{7/3}$	"
3	n^2	"

Some log factors ignores

Spanners



Given an **arbitrary** dense graph, can we always find a relatively **sparse subgraph** that approximates **all** distances fairly well?

Spanners [PU'89,PS'89]

Let $G=(V,E)$ be a weighted undirected graph.

A subgraph $G'=(V,E')$ of G is said to be a t-spanner of G iff $\delta_{G'}(u,v) \le t\,\delta_G(u,v)$ for every u,v in V.

Theorem:

Every weighted undirected graph has a $(2k-1)$-spanner of size $O(n^{1+1/k})$. [ADDJS '93]

Furthermore, such spanners can be constructed deterministically in linear time. [BS '04] [TZ '04]

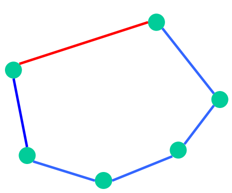
The size-stretch trade-off is essentially optimal.

(Assuming there are graphs with $\Omega(n^{1+1/k})$ edges of girth $2k+2$, as conjectured by Erdös and others.)

Existence Proof / Construction Algorithm

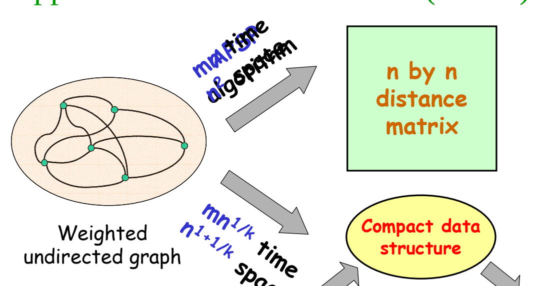
[Althöfer, Das, Dobkin, Joseph, Soares '93]

- Consider the edges of the graph in non-decreasing order of weight.
- Add each edge to the spanner if it does not close a cycle of size at most $2k$.
- The resulting graph is a $(2k-1)$-spanner.
- The resulting graph does not contain a cycle of size at most $2k$.
- Hence the number of edges in it is at most $n^{1+1/k}$.



If |cycle|≤2k, then red edge can be removed.

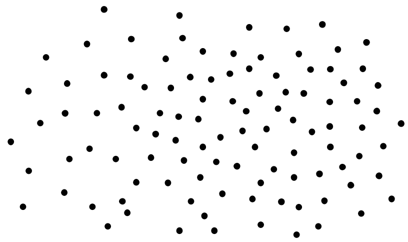
Approximate Distance Oracles (TZ'01)



Stretch-Space tradeoff is essentially optimal!

Approximate Distance Oracles [TZ'01]

A hierarchy of centers

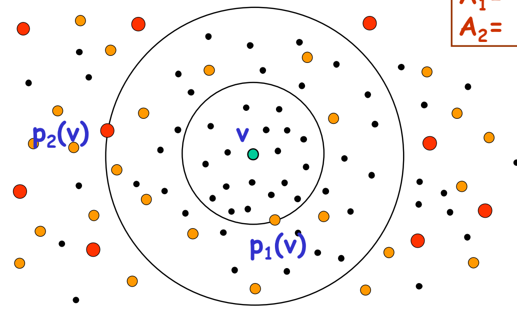


$A_0 \leftarrow V$; $A_k \leftarrow \varnothing$;

$A_i \leftarrow sample(A_{i-1}, n^{-1/k})$;

Bunches

$A_0 =$ •
$A_1 =$ •
$A_2 =$ •



$$B(v) \leftarrow \bigcup_i \{ w \in A_i - A_{i+1} \mid \delta(w,v) < \delta(A_{i+1},v) \}$$

Lemma: $E[|B(v)|] \leq kn^{1/k}$

Proof: $|B(v) \cap A_i|$ is stochastically dominated by a geometric random variable with parameter $p=n^{-1/k}$.

The data structure

Keep for every vertex $v \in V$:

- The centers $p_1(v), p_2(v), \ldots, p_{k-1}(v)$
- A **hash table** holding $B(v)$

For every $w \in V$, we can check, in **constant time**, whether $w \in B(v)$, and if so, what is $\delta(v,w)$.

Query answering algorithm

Algorithm $dist_k(u,v)$

$w \leftarrow u$, $i \leftarrow 0$

while $w \notin B(v)$

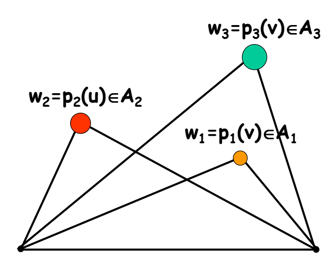
{ $i \leftarrow i+1$

$(u,v) \leftarrow (v,u)$

$w \leftarrow p_i(u)$ }

return $\delta(u,w) + \delta(w,v)$

Query answering algorithm



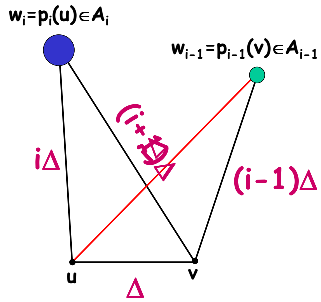
Analysis

Claim 1:

$\delta(u,w_i) \le i\Delta$, i even
$\delta(v,w_i) \le i\Delta$, i odd

Claim 2:

$\delta(u,w_i) + \delta(w_i,v)$
$\le (2i+1)\Delta$
$\le (2k-1)\Delta$

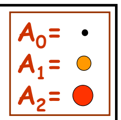


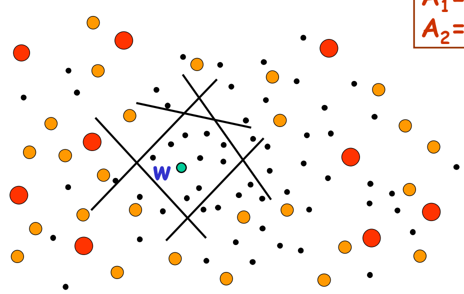
Where are the spanners?

Define clusters, the “dual” of bunches.

For every $u \in V$, put in the spanner a tree of shortest paths from u to all the vertices in the cluster of u.

Clusters





$$C(w) \leftarrow \{v \in V \mid \delta(w,v) < \delta(A_{i+1},v)\} \quad , \quad w \in A_i - A_{i+1}$$

Bunches and clusters

$$w \in B(v) \iff v \in C(w)$$

$$C(w) \leftarrow \{v \in V \mid \delta(w,v) < \delta(A_{i+1},v)\} \quad , \quad \text{if } w \in A_i - A_{i+1}$$

$$B(v) \leftarrow \bigcup_i \{w \in A_i - A_{i+1} \mid \delta(w,v) < \delta(A_{i+1},v)\}$$

Additive Spanners

Let $G=(V,E)$ be a unweighted undirected graph.

A subgraph $G'=(V,E')$ of G is said to be an additive t-spanner if G iff $\delta_{G'}(u,v) \le \delta_G(u,v) + t$ for every $u,v \in V$.

Theorem: Every unweighted undirected graph has an additive 2-spanner of size $O(n^{3/2})$. [ACIM '96] [DHZ '96]

Theorem: Every unweighted undirected graph has an additive 6-spanner of size $O(n^{4/3})$. [BKMP '04]

Major open problem

Do all graphs have additive spanners with only $O(n^{1+\varepsilon})$ edges, for every $\varepsilon>0$?

Spanners with sublinear surplus

Theorem:

For every $k>1$, every undirected graph $G=(V,E)$ on n vertices has a subgraph $G'=(V,E')$ with $O(n^{1+1/k})$ edges such that for every $u,v \in V$, if $\delta_G(u,v)=d$, then $\delta_{G'}(u,v)=d+O(d^{1-1/(k-1)})$.

$d \Rightarrow d+O(d^{1-1/(k-1)})$

Extends and simplifies a result of Elkin and Peleg (2001)

All sorts of spanners

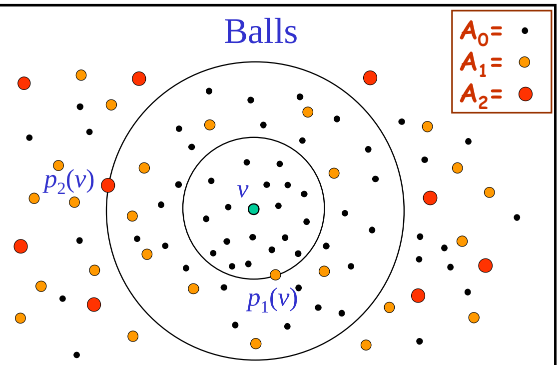
A subgraph $G'=(V,E')$ of G is said to be a functional f-spanner if G iff $\delta_{G'}(u,v) \leq f(\delta_G(u,v))$ for every $u,v \in V$.

size	f(d)	reference
$n^{1+1/k}$	$(2k-1)d$	[ADDJS '93]
$n^{3/2}$	$d+2$	[ACIM '96] [DHZ '96]
$n^{4/3}$	$d+6$	[BKMP '04]
$\beta n^{1+\delta}$	$(1+\varepsilon)d+\beta(\varepsilon,\delta)$	[EP '01]
$n^{1+1/k}$	$d+O(d^{1-1/(k-1)})$	[TZ '05]

The construction of the approximate distance oracles, when applied to unweighted graphs, produces spanners with sublinear surplus!

We present a slightly modified construction with a slightly simpler analysis.

Balls



$$Ball(u) = \{v \in V \mid \delta(u,v) < \delta(u, A_{i+1})\} \ , \ u \in A_i - A_{i+1}$$
$$Ball[u] = Ball(u) \cup \{p_{i+1}(u)\} \ , \ u \in A_i - A_{i+1}$$

The original construction

Select a hierarchy of centers $A_0 \supset A_1 \supset \ldots \supset A_{k-1}$.

For every $u \in V$, add to the spanner a shortest paths tree of Clust(u).

The modified construction

Select a hierarchy of centers $A_0 \supset A_1 \supset \ldots \supset A_{k-1}$.

For every $u \in V$, add to the spanner a shortest paths tree of Ball(u).

Spanners with sublinear surplus

Select a hierarchy of centers $A_0 \supset A_1 \supset \ldots \supset A_{k-1}$.

For every $u \in V$, add to the spanner a shortest paths tree of Ball(u).

The path-finding strategy

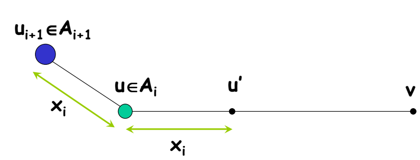
Suppose we are at $u \in A_i$ and want to go to v.

Let Δ be an integer parameter.

If the first $x_i = \Delta^i - \Delta^{i-1}$ edges of a shortest path from u to v are in the spanner, then use them.

Otherwise, head for the $(i+1)$-center u_{i+1} nearest to u.

► The distance to u_{i+1} is at most x_i. (As $u' \notin Ball(u)$.)

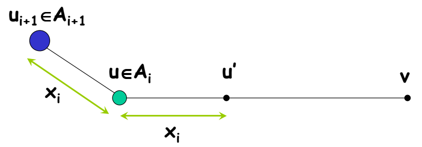


The path-finding strategy

We either reach v, or at least make $x_i=\Delta^i-\Delta^{i-1}$ steps in the right direction.

Or, make at most $x_i=\Delta^i-\Delta^{i-1}$ steps, possibly in a wrong direction, but reach a center of level $i+1$.

If $i=k-1$, we will be able to reach v.

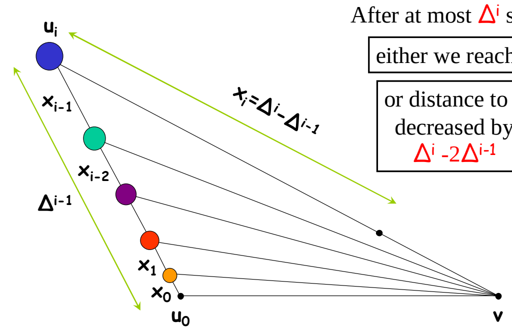


The path-finding strategy

After at most Δ^i steps:

either we reach v

or distance to v decreased by $\Delta^i-2\Delta^{i-1}$



The path-finding strategy

After at most Δ^i steps:

either we reach v	⇒	Surplus $2\Delta^{i-1}$
or distance to v decreased by $\Delta^i-2\Delta^{i-1}$	⇒	Stretch $\frac{\Delta^i}{\Delta^i-2\Delta^{i-1}} = 1+\frac{2}{\Delta-2}$

The surplus is incurred only once!

$$\delta'(u,v) \le (1+\frac{2}{\Delta-2})\cdot\delta(u,v) + 2\Delta^{k-2}$$

Sublinear surplus

$$\delta'(u,v) \le (1+\frac{2}{\Delta-2})\cdot\delta(u,v) + 2\Delta^{k-2}$$

$$\delta(u,v)=d \quad , \quad \Delta=\lceil d^{1/(k-1)}+2\rceil$$

⇓

$$\delta'(u,v) \le d+O(d^{1-\frac{1}{k-1}})$$