Welcome back...

Welcome back...

Metric spaces.

A metric space X, $d(i,j)$ where $d(i,j) \leq d(i,k) + d(k,l)$ and $d(i,j) = d(j,i)$

Metric spaces.

A metric space X, $d(i,j)$ where $d(i,j) \leq d(i,k) + d(k,l)$ and $d(i,j) = d(j,i)$

Which are metric spaces?

(A) X from R^d and $d(\cdot,\cdot)$ is Euclidean distance.

(B) X from R^d and $d(\cdot,\cdot)$ is squared Euclidean distance.

(C) X- vertices in graph, $d(i,j)$ is shortest path distances in graph.

(D) X is a set of vectors and $d(u,v)$ is $u \cdot v$.

Metric spaces.

A metric space X, $d(i,j)$ where $d(i,j) \leq d(i,k) + d(k,l)$ and $d(i,j) = d(j,i)$

Which are metric spaces?

(A) X from R^d and $d(\cdot,\cdot)$ is Euclidean distance.

(B) X from R^d and $d(\cdot,\cdot)$ is squared Euclidean distance.

(C) X- vertices in graph, $d(i,j)$ is shortest path distances in graph.

(D) X is a set of vectors and $d(u,v)$ is $u \cdot v$.

Metric spaces.

A metric space X, $d(i,j)$ where $d(i,j) \leq d(i,k) + d(k,l)$ and $d(i,j) = d(j,i)$

Which are metric spaces?

(A) X from R^d and $d(\cdot,\cdot)$ is Euclidean distance.

(B) X from R^d and $d(\cdot,\cdot)$ is squared Euclidean distance.

(C) X- vertices in graph, $d(i,j)$ is shortest path distances in graph.

(D) X is a set of vectors and $d(u,v)$ is $u \cdot v$.

Input to TSP, facility location, some layout problems, ..., metric labelling.

Metric spaces.

A metric space X, $d(i,j)$ where $d(i,j) \leq d(i,k) + d(k,l)$ and $d(i,j) = d(j,i)$

Which are metric spaces?

(A) X from R^d and $d(\cdot,\cdot)$ is Euclidean distance.

(B) X from R^d and $d(\cdot,\cdot)$ is squared Euclidean distance.

(C) X- vertices in graph, $d(i,j)$ is shortest path distances in graph.

(D) X is a set of vectors and $d(u,v)$ is $u \cdot v$.

Input to TSP, facility location, some layout problems, ..., metric labelling.

Hard problems.

Metric spaces.

A metric space X, $d(i,j)$ where $d(i,j) \leq d(i,k) + d(k,l)$ and $d(i,j) = d(j,i)$

Which are metric spaces?

(A) X from R^d and $d(\cdot,\cdot)$ is Euclidean distance.

(B) X from R^d and $d(\cdot,\cdot)$ is squared Euclidean distance.

(C) X- vertices in graph, $d(i,j)$ is shortest path distances in graph.

(D) X is a set of vectors and $d(u,v)$ is $u \cdot v$.

Input to TSP, facility location, some layout problems, ..., metric labelling.

Hard problems. Easier to solve on trees.

Metric spaces.

A metric space X, $d(i,j)$ where $d(i,j) \leq d(i,k) + d(k,l)$ and $d(i,j) = d(j,i)$

Which are metric spaces?

(A) X from R^d and $d(\cdot,\cdot)$ is Euclidean distance.

(B) X from R^d and $d(\cdot,\cdot)$ is squared Euclidean distance.

(C) X- vertices in graph, $d(i,j)$ is shortest path distances in graph.

(D) X is a set of vectors and $d(u,v)$ is $u \cdot v$.

Input to TSP, facility location, some layout problems, ..., metric labelling.

Hard problems. Easier to solve on trees. Dynamic programming on trees.

Metric spaces.

A metric space X, $d(i,j)$ where $d(i,j) \leq d(i,k) + d(k,l)$ and $d(i,j) = d(j,i)$

Which are metric spaces?

(A) X from R^d and $d(\cdot,\cdot)$ is Euclidean distance.

(B) X from R^d and $d(\cdot,\cdot)$ is squared Euclidean distance.

(C) X- vertices in graph, $d(i,j)$ is shortest path distances in graph.

(D) X is a set of vectors and $d(u,v)$ is $u \cdot v$.

Input to TSP, facility location, some layout problems, ..., metric labelling.

Hard problems. Easier to solve on trees. Dynamic programming on trees.

Approximate metric on trees?

Approximate metric using a tree.

Tree metric:

Approximate metric using a tree.

Tree metric:

X is nodes of tree with edge weights

$d_T(i,j)$ shortest path metric on tree.

Approximate metric using a tree.

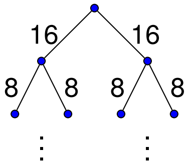
Tree metric:
  X is nodes of tree with edge weights
  $d_T(i,j)$ shortest path metric on tree.

Hierarchically well separated tree metric:

Approximate metric using a tree.

Tree metric:
  X is nodes of tree with edge weights
  $d_T(i,j)$ shortest path metric on tree.

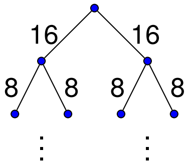
Hierarchically well separated tree metric:
  Tree weights are geometrically decreasing.



Approximate metric using a tree.

Tree metric:
  X is nodes of tree with edge weights
  $d_T(i,j)$ shortest path metric on tree.

Hierarchically well separated tree metric:
  Tree weights are geometrically decreasing.

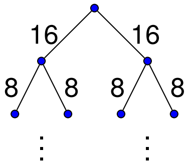


Map X into tree.

Approximate metric using a tree.

Tree metric:
  X is nodes of tree with edge weights
  $d_T(i,j)$ shortest path metric on tree.

Hierarchically well separated tree metric:
  Tree weights are geometrically decreasing.

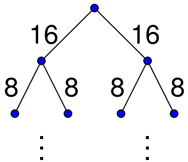


Map X into tree.
 (i) No distance shrinks. (dominating)

Approximate metric using a tree.

Tree metric:
X is nodes of tree with edge weights
$d_T(i,j)$ shortest path metric on tree.

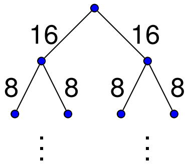
Hierarchically well separated tree metric:
Tree weights are geometrically decreasing.



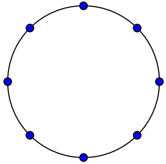
Map X into tree.
(i) No distance shrinks. (dominating)
(ii) Every distance stretches $\leq \alpha$

Approximate metric using a tree.

Tree metric:
  X is nodes of tree with edge weights
  $d_T(i,j)$ shortest path metric on tree.



Hierarchically well separated tree metric:
  Tree weights are geometrically decreasing.



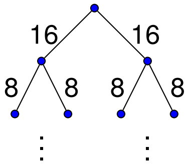
Map X into tree.
(i) No distance shrinks. (dominating)
(ii) Every distance stretches $\leq \alpha$

Map metric onto tree?

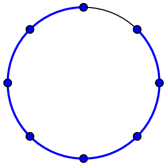
Approximate metric using a tree.

Tree metric:
X is nodes of tree with edge weights
$d_T(i,j)$ shortest path metric on tree.

Hierarchically well separated tree metric:
Tree weights are geometrically decreasing.



Map X into tree.
(i) No distance shrinks. (dominating)
(ii) Every distance stretches $\leq \alpha$

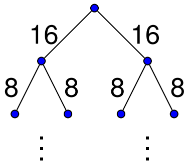


Map metric onto tree?

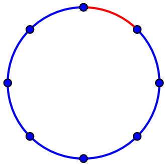
Approximate metric using a tree.

Tree metric:
X is nodes of tree with edge weights
$d_T(i,j)$ shortest path metric on tree.

Hierarchically well separated tree metric:
Tree weights are geometrically decreasing.



Map X into tree.
(i) No distance shrinks. (dominating)
(ii) Every distance stretches $\leq \alpha$

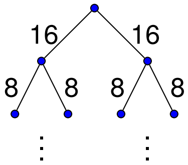


Distance 1 goes to $n-1$!

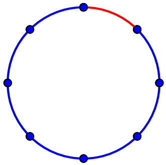
Map metric onto tree?

Approximate metric using a tree.

Tree metric:
  X is nodes of tree with edge weights
  $d_T(i,j)$ shortest path metric on tree.



Hierarchically well separated tree metric:
  Tree weights are geometrically decreasing.



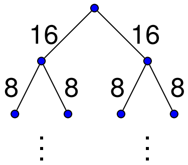
Map X into tree.
 (i) No distance shrinks. (dominating)
 (ii) Every distance stretches $\leq \alpha$

Distance 1 goes to $n-1$!
 Bummer.

Map metric onto tree?

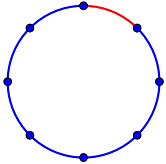
Approximate metric using a tree.

Tree metric:
  X is nodes of tree with edge weights
  $d_T(i,j)$ shortest path metric on tree.



Hierarchically well separated tree metric:
  Tree weights are geometrically decreasing.

Probabilistic Tree embedding.
  Map X into tree.
  (i) No distance shrinks. (dominating)
  (ii) Every distance stretches $\leq \alpha$
    in expectation.



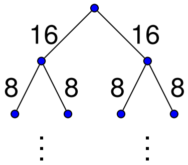
Distance 1 goes to $n-1$!
  Bummer.

Map metric onto tree?

Fix it up chappie!

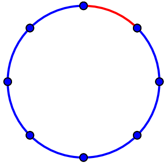
Approximate metric using a tree.

Tree metric:
  X is nodes of tree with edge weights
  $d_T(i,j)$ shortest path metric on tree.



Hierarchically well separated tree metric:
  Tree weights are geometrically decreasing.

Probabilistic Tree embedding.
  Map X into tree.
   (i) No distance shrinks. (dominating)
   (ii) Every distance stretches $\leq \alpha$
     in expectation.



Distance 1 goes to $n-1$!
  Bummer.

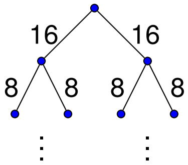
Map metric onto tree?

Fix it up chappie!

For cycle, remove a random edge

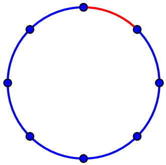
Approximate metric using a tree.

Tree metric:
  X is nodes of tree with edge weights
  $d_T(i,j)$ shortest path metric on tree.



Hierarchically well separated tree metric:
  Tree weights are geometrically decreasing.

Probabilistic Tree embedding.
  Map X into tree.
   (i) No distance shrinks. (dominating)
   (ii) Every distance stretches $\leq \alpha$
     in expectation.



Distance 1 goes to $n-1$!
  Bummer.

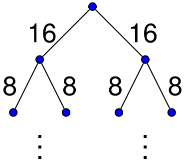
Map metric onto tree?

Fix it up chappie!

For cycle, remove a random edge get a tree.

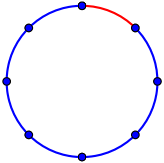
Approximate metric using a tree.

Tree metric:
X is nodes of tree with edge weights
$d_T(i,j)$ shortest path metric on tree.



Hierarchically well separated tree metric:
Tree weights are geometrically decreasing.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks. (dominating)
(ii) Every distance stretches $\leq \alpha$
in expectation.



Distance 1 goes to $n-1$!
Bummer.

Map metric onto tree?

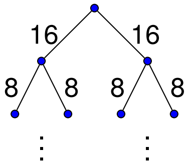
Fix it up chappie!

For cycle, remove a random edge get a tree.

Stretch of edge: $\frac{n-1}{n} \times 1$

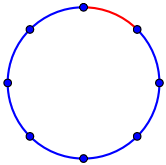
Approximate metric using a tree.

Tree metric:
X is nodes of tree with edge weights
$d_T(i,j)$ shortest path metric on tree.



Hierarchically well separated tree metric:
Tree weights are geometrically decreasing.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks. (dominating)
(ii) Every distance stretches $\leq \alpha$
in expectation.



Distance 1 goes to $n-1$!
Bummer.

Map metric onto tree?

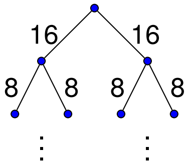
Fix it up chappie!

For cycle, remove a random edge get a tree.

Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1)$

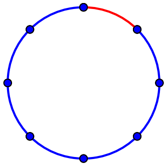
Approximate metric using a tree.

Tree metric:
X is nodes of tree with edge weights
$d_T(i,j)$ shortest path metric on tree.



Hierarchically well separated tree metric:
Tree weights are geometrically decreasing.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks. (dominating)
(ii) Every distance stretches $\leq \alpha$
in expectation.



Distance 1 goes to $n-1$!
Bummer.

Map metric onto tree?

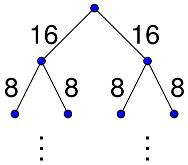
Fix it up chappie!

For cycle, remove a random edge get a tree.

Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1) \approx$

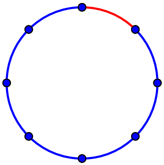
Approximate metric using a tree.

Tree metric:
 X is nodes of tree with edge weights
 $d_T(i,j)$ shortest path metric on tree.



Hierarchically well separated tree metric:
 Tree weights are geometrically decreasing.

Probabilistic Tree embedding.
 Map X into tree.
 (i) No distance shrinks. (dominating)
 (ii) Every distance stretches $\leq \alpha$
 in expectation.



Distance 1 goes to $n-1$!
 Bummer.

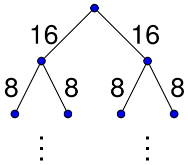
Map metric onto tree?

Fix it up chappie!

For cycle, remove a random edge get a tree.

Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1) \approx 2$

Approximate metric using a tree.

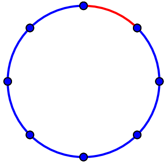


Tree metric:
X is nodes of tree with edge weights
$d_T(i,j)$ shortest path metric on tree.

Hierarchically well separated tree metric:
Tree weights are geometrically decreasing.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks. (dominating)
(ii) Every distance stretches $\leq \alpha$
in expectation.

Map metric onto tree?



Distance 1 goes to $n-1$!
Bummer.

Fix it up chappie!

For cycle, remove a random edge get a tree.

Stretch of edge: $\frac{n-1}{n} \times 1 + \frac{1}{n} \times (n-1) \approx 2$

General metrics?

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.

Probabilistic Tree embedding.

Probabilistic Tree embedding.

Map X into tree.

(i) No distance shrinks (dominating).

Probabilistic Tree embedding.

Probabilistic Tree embedding.

Map X into tree.

(i) No distance shrinks (dominating).

(ii) Every distance stretches $\leq \alpha$ in expecation.

Probabilistic Tree embedding.

Probabilistic Tree embedding.

Map X into tree.

(i) No distance shrinks (dominating).

(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).

Probabilistic Tree embedding.

Probabilistic Tree embedding.

Map X into tree.

(i) No distance shrinks (dominating).

(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).

Elements of X are leaves of tree.

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks (dominating).
(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).
Elements of X are leaves of tree.

On Tuesday: **use** spanning tree for graphical metrics.

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks (dominating).
(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).
Elements of X are leaves of tree.

On Tuesday: **use** spanning tree for graphical metrics.

The Idea:
HST $\equiv$ recursive decomposition of metric space.

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks (dominating).
(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).
Elements of X are leaves of tree.

On Tuesday: **use** spanning tree for graphical metrics.

The Idea:
HST $\equiv$ recursive decomposition of metric space.

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks (dominating).
(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).
Elements of X are leaves of tree.

On Tuesday: **use** spanning tree for graphical metrics.

The Idea:
HST $\equiv$ recursive decomposition of metric space.

Decompose space by diameter $\approx \Delta$ balls.

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks (dominating).
(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).
Elements of X are leaves of tree.

On Tuesday: **use** spanning tree for graphical metrics.

The Idea:
HST $\equiv$ recursive decomposition of metric space.

Decompose space by diameter $\approx \Delta$ balls.
Recurse on each ball for $\Delta/2$.

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks (dominating).
(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).
Elements of X are leaves of tree.

On Tuesday: **use** spanning tree for graphical metrics.

The Idea:
HST $\equiv$ recursive decomposition of metric space.

Decompose space by diameter $\approx \Delta$ balls.
Recurse on each ball for $\Delta/2$.

Use randomness in

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks (dominating).
(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).
Elements of X are leaves of tree.

On Tuesday: **use** spanning tree for graphical metrics.

The Idea:
HST $\equiv$ recursive decomposition of metric space.

Decompose space by diameter $\approx \Delta$ balls.
Recurse on each ball for $\Delta/2$.

Use randomness in
selection of ball centers.

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks (dominating).
(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).
Elements of X are leaves of tree.

On Tuesday: **use** spanning tree for graphical metrics.

The Idea:
HST $\equiv$ recursive decomposition of metric space.

Decompose space by diameter $\approx \Delta$ balls.
Recurse on each ball for $\Delta/2$.

Use randomness in
selection of ball centers.
the $\approx$ diameter of the balls.

Probabilistic Tree embedding.

Probabilistic Tree embedding.
Map X into tree.
(i) No distance shrinks (dominating).
(ii) Every distance stretches $\leq \alpha$ in expecation.

Today: the tree will be Hierarchically well-separated (HST).
Elements of X are leaves of tree.

On Tuesday: **use** spanning tree for graphical metrics.

The Idea:
HST $\equiv$ recursive decomposition of metric space.

Decompose space by diameter $\approx \Delta$ balls.
Recurse on each ball for $\Delta/2$.

Use randomness in
selection of ball centers.
the $\approx$ diameter of the balls.

Algorithm

Algorithm: (X, d), $\text{diam}(X) \leq D$, $|X| = n$, $d(i, j) \geq 1$

Algorithm

Algorithm: (X, d), $\text{diam}(X) \le D$, $|X| = n$, $d(i, j) \ge 1$

1. π – random permutation of X.

Algorithm

Algorithm: (X, d), $\text{diam}(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

Algorithm

Algorithm: (X, d), diam$(X) \leq D$, $|X| = n$, $d(i, j) \geq 1$
1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.
   def subtree(S,Δ):

Algorithm

Algorithm: (X, d), diam$(X) \leq D$, $|X| = n$, $d(i, j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
 def subtree(S,Δ):
  T = []
```

Algorithm

Algorithm: (X, d), diam$(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
 def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
```

Algorithm

Algorithm: (X, d), diam$(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
 def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
```

Algorithm

Algorithm: (X, d), diam$(X) \leq D$, $|X| = n$, $d(i, j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
    if i ∈ S
```

Algorithm

Algorithm: (X, d), diam$(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
    if i ∈ S
      B = ball(i, βΔ)
```

Algorithm

Algorithm: (X, d), $\text{diam}(X) \leq D$, $|X| = n$, $d(i, j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
    if i ∈ S
      B = ball(i, βΔ) ; S = S/B
```

Algorithm

Algorithm: (X, d), diam$(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
    if i ∈ S
      B = ball(i, βΔ) ; S = S/B
      T.append(B)
```

Algorithm

Algorithm: (X, d), $\text{diam}(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
    if i ∈ S
      B = ball(i, βΔ) ; S = S/B
      T.append(B)
  return map (λ x: subtree(x,Δ/2), T);
```

Algorithm

Algorithm: (X, d), diam$(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

```
1. π – random permutation of X.
2. Choose β in [1/4, 1/2].
 def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
   if i ∈ S
    B = ball(i, βΔ) ; S = S/B
    T.append(B)
  return map (λ x: subtree(x,Δ/2), T);
3. subtree(X,Δ)
```

Algorithm

Algorithm: (X, d), $\text{diam}(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
 def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
    if i ∈ S
      B = ball(i, βΔ) ; S = S/B
      T.append(B)
  return map (λ x: subtree(x,Δ/2), T);
```

3. subtree(X, Δ)

Tree has internal node for each level of call.

Algorithm

Algorithm: (X, d), $\text{diam}(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
 def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
   if i ∈ S
     B = ball(i, βΔ) ; S = S/B
     T.append(B)
  return map (λ x: subtree(x,Δ/2), T);
```

3. subtree(X, Δ)

Tree has internal node for each level of call. Tree edges have weight $\Delta/2$ to children.

Algorithm

Algorithm: (X, d), diam$(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
 def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
   if i ∈ S
     B = ball(i, βΔ) ; S = S/B
     T.append(B)
  return map (λ x: subtree(x,Δ/2), T);
```

3. subtree(X, Δ)

Tree has internal node for each level of call. Tree edges have weight $\Delta/2$ to children.

Claim 1: $d_T(x,y) \geq d(x,y)$.

Algorithm

Algorithm: (X,d), $\text{diam}(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
 def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
   if i ∈ S
     B = ball(i, βΔ) ; S = S/B
     T.append(B)
  return map (λ x: subtree(x,Δ/2), T);
```

3. subtree(X,Δ)

Tree has internal node for each level of call. Tree edges have weight $\Delta/2$ to children.

Claim 1: $d_T(x,y) \geq d(x,y)$.

$d(x,y)$ are in different sets at level $\Delta \leq d(x,y)$.

Algorithm

Algorithm: (X, d), $\text{diam}(X) \leq D$, $|X| = n$, $d(i,j) \geq 1$

1. π – random permutation of X.
2. Choose β in $[\frac{1}{4}, \frac{1}{2}]$.

```
 def subtree(S,Δ):
  T = []
  if Δ < 1 return [S]
  foreach i in π:
    if i ∈ S
      B = ball(i, βΔ) ; S = S/B
      T.append(B)
  return map (λ x: subtree(x,Δ/2), T);
```

3. subtree(X, Δ)

Tree has internal node for each level of call. Tree edges have weight $\Delta/2$ to children.

Claim 1: $d_T(x,y) \geq d(x,y)$.

$d(x,y)$ are in different sets at level $\Delta \leq d(x,y)$.

$\rightarrow d(x,y) \geq \Delta$

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n) d(x,y)$.

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n)d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$.

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n) d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n) d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

$Pr[\text{cut at level}\Delta]$?

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n) d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n) d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\rightarrow$ expected length is $\sum_{\Delta=D/2^i} (2\Delta) \frac{d(x,y)}{\Delta} = 2d(x,y)$.

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n) d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\rightarrow$ expected length is $\sum_{\Delta = D/2^i} (2\Delta) \frac{d(x,y)}{\Delta} = 2d(x,y)$.

Why should it be $\frac{d(x,y)}{\Delta}$?

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n) d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\rightarrow$ expected length is $\sum_{\Delta = D/2^i} (2\Delta) \frac{d(x,y)}{\Delta} = 2d(x,y)$.

Why should it be $\frac{d(x,y)}{\Delta}$?
smaller the edge the less likely to be on edge of ball.

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n) d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\rightarrow$ expected length is $\sum_{\Delta=D/2^i}(2\Delta)\frac{d(x,y)}{\Delta} = 2d(x,y)$.

Why should it be $\frac{d(x,y)}{\Delta}$?

smaller the edge the less likely to be on edge of ball.

larger the delta, more room inside ball.

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n)d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\rightarrow$ expected length is $\sum_{\Delta=D/2^i}(2\Delta)\frac{d(x,y)}{\Delta} = 2d(x,y)$.

Why should it be $\frac{d(x,y)}{\Delta}$?
smaller the edge the less likely to be on edge of ball.
larger the delta, more room inside ball.
random diameter jiggles edge of ball.

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n)d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\rightarrow$ expected length is $\sum_{\Delta=D/2^i}(2\Delta)\frac{d(x,y)}{\Delta} = 2d(x,y)$.

Why should it be $\frac{d(x,y)}{\Delta}$?

smaller the edge the less likely to be on edge of ball.

larger the delta, more room inside ball.

random diameter jiggles edge of ball.

$\rightarrow Pr[x,y$ cut by ball$|x$ in ball$] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n)d(x,y)$.

Cut at level $\Delta \to d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\to$ expected length is $\sum_{\Delta=D/2^i}(2\Delta)\frac{d(x,y)}{\Delta} = 2d(x,y)$.

Why should it be $\frac{d(x,y)}{\Delta}$?
smaller the edge the less likely to be on edge of ball.
larger the delta, more room inside ball.
random diameter jiggles edge of ball.

$\to Pr[x,y$ cut by ball$|x$ in ball$] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$

The problem?

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n) d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\rightarrow$ expected length is $\sum_{\Delta=D/2^i}(2\Delta)\frac{d(x,y)}{\Delta} = 2d(x,y)$.

Why should it be $\frac{d(x,y)}{\Delta}$?
smaller the edge the less likely to be on edge of ball.
larger the delta, more room inside ball.
random diameter jiggles edge of ball.

$\rightarrow Pr[x, y$ cut by ball$|x$ in ball$] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$

The problem?
Could be cut be many different balls.

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n)d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

Pr[cut at levelΔ]?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\rightarrow$ expected length is $\sum_{\Delta=D/2^i}(2\Delta)\frac{d(x,y)}{\Delta} = 2d(x,y)$.

Why should it be $\frac{d(x,y)}{\Delta}$?
smaller the edge the less likely to be on edge of ball.
larger the delta, more room inside ball.
random diameter jiggles edge of ball.

$\rightarrow Pr[x,y \text{ cut by ball}|x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$

The problem?
Could be cut be many different balls.
For each probability is good, but could be hit by many.

Analysis: idea

Claim: $E[d_T(x,y)] = O(\log n)d(x,y)$.

Cut at level $\Delta \rightarrow d_T(x,y) \approx 2\Delta$. (Level of subtree call.)

$Pr[\text{cut at level}\Delta]$?

Would like it to be $\frac{d(x,y)}{\Delta}$.

$\rightarrow$ expected length is $\sum_{\Delta=D/2^i}(2\Delta)\frac{d(x,y)}{\Delta} = 2d(x,y)$.

Why should it be $\frac{d(x,y)}{\Delta}$?

smaller the edge the less likely to be on edge of ball.

larger the delta, more room inside ball.

random diameter jiggles edge of ball.

$\rightarrow Pr[x,y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$

The problem?

Could be cut be many different balls.

For each probability is good, but could be hit by many.

random permutation to deal with this

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$

(Only consider cut by x, factor 2 loss.)

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$

(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$

(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.

Renumber nodes in order of distance from x.

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j, x) \in [\Delta/4, \Delta/2]$,

Analysis: (x,y)

Have $Pr[x,y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j,x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

Analysis: (x,y)

Have $Pr[x,y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j,x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x,y) if..

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j, x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x, y) if..
$d(j, x) \leq \beta\Delta$

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j, x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x, y) if..
$d(j, x) \leq \beta\Delta$ and $\beta\Delta \leq d(j, y)$

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j, x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x, y) if..
$d(j, x) \leq \beta\Delta$ and $\beta\Delta \leq d(j, y) \leq d(j, x) + d(x, y)$

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j,x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x, y) if..
$d(j,x) \leq \beta\Delta$ and $\beta\Delta \leq d(j,y) \leq d(j,x) + d(x,y)$
$\rightarrow \beta\Delta \in [d[j,x], d(j,x) + d(x,y)]$.

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j, x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x, y) if..
$d(j, x) \leq \beta\Delta$ and $\beta\Delta \leq d(j, y) \leq d(j, x) + d(x, y)$
$\rightarrow \beta\Delta \in [d[j, x], d(j, x) + d(x, y)]$.
occurs with prob. $\frac{d(x,y)}{\Delta/4} = \frac{4d(x,y)}{\Delta}$.

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j, x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x, y) if..
$d(j, x) \leq \beta\Delta$ and $\beta\Delta \leq d(j, y) \leq d(j, x) + d(x, y)$
$\rightarrow \beta\Delta \in [d[j, x], d(j, x) + d(x, y)]$.
occurs with prob. $\frac{d(x,y)}{\Delta/4} = \frac{4d(x,y)}{\Delta}$.

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j, x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x, y) if..
$d(j,x) \leq \beta\Delta$ and $\beta\Delta \leq d(j,y) \leq d(j,x) + d(x,y)$
$\rightarrow \beta\Delta \in [d[j,x], d(j,x) + d(x,y)]$.
occurs with prob. $\frac{d(x,y)}{\Delta/4} = \frac{4d(x,y)}{\Delta}$.

And j must be before any $i < j$ in π

Analysis: (x,y)

Have $Pr[x,y \text{ cut by ball}|x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j,x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x,y) if..
$d(j,x) \leq \beta\Delta$ and $\beta\Delta \leq d(j,y) \leq d(j,x) + d(x,y)$
$\rightarrow \beta\Delta \in [d[j,x], d(j,x) + d(x,y)]$.
occurs with prob. $\frac{d(x,y)}{\Delta/4} = \frac{4d(x,y)}{\Delta}$.

And j must be before any $i < j$ in $\pi \rightarrow$ prob is $\frac{1}{j}$

Analysis: (x,y)

Have $Pr[x,y \text{ cut by ball}|x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j,x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x,y) if..
$d(j,x) \leq \beta\Delta$ and $\beta\Delta \leq d(j,y) \leq d(j,x) + d(x,y)$
$\rightarrow \beta\Delta \in [d[j,x], d(j,x) + d(x,y)]$.
occurs with prob. $\frac{d(x,y)}{\Delta/4} = \frac{4d(x,y)}{\Delta}$.

And j must be before any $i < j$ in $\pi \rightarrow$ prob is $\frac{1}{j}$

$\rightarrow Pr[j \text{ cuts } (x,y)] \leq \left(\frac{1}{j}\right) \frac{4d(x,y)}{\Delta}$

Analysis: (x, y)

Have $Pr[x, y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j, x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x, y) if..
$d(j, x) \leq \beta\Delta$ and $\beta\Delta \leq d(j, y) \leq d(j, x) + d(x, y)$
$\rightarrow \beta\Delta \in [d[j, x], d(j, x) + d(x, y)]$.
occurs with prob. $\frac{d(x,y)}{\Delta/4} = \frac{4d(x,y)}{\Delta}$.

And j must be before any $i < j$ in $\pi \rightarrow$ prob is $\frac{1}{j}$

$\rightarrow Pr[j \text{ cuts } (x, y)] \leq \left(\frac{1}{j}\right) \frac{4d(x,y)}{\Delta}$

$d_T(x, y)$ if cut level Δ is 2Δ.

Analysis: (x,y)

Have $Pr[x,y \text{ cut by ball} | x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j,x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x,y) if..
$d(j,x) \leq \beta\Delta$ and $\beta\Delta \leq d(j,y) \leq d(j,x) + d(x,y)$
$\rightarrow \beta\Delta \in [d[j,x], d(j,x) + d(x,y)]$.
occurs with prob. $\frac{d(x,y)}{\Delta/4} = \frac{4d(x,y)}{\Delta}$.

And j must be before any $i < j$ in $\pi \rightarrow$ prob is $\frac{1}{j}$

$\rightarrow Pr[j \text{ cuts } (x,y)] \leq \left(\frac{1}{j}\right) \frac{4d(x,y)}{\Delta}$

$d_T(x,y)$ if cut level Δ is 2Δ.
$\rightarrow E[d_T(x,y)] = \sum_{\Delta = \frac{D}{2^i}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y)$

Analysis: (x,y)

Have $Pr[x,y \text{ cut by ball}|x \text{ in ball}] \approx \frac{d(x,y)}{\beta\Delta} \leq 4\Delta$
(Only consider cut by x, factor 2 loss.)

At level Δ

At some point x is in some Δ level ball.
Renumber nodes in order of distance from x.

Can only in ball for j, where $d(j,x) \in [\Delta/4, \Delta/2]$,
Call this set X_Δ.

If $j \in X_\Delta$ cuts (x,y) if..
$d(j,x) \leq \beta\Delta$ and $\beta\Delta \leq d(j,y) \leq d(j,x) + d(x,y)$
$\rightarrow \beta\Delta \in [d[j,x], d(j,x) + d(x,y)]$.
occurs with prob. $\frac{d(x,y)}{\Delta/4} = \frac{4d(x,y)}{\Delta}$.

And j must be before any $i < j$ in $\pi \rightarrow$ prob is $\frac{1}{j}$

$\rightarrow Pr[j \text{ cuts } (x,y)] \leq \left(\frac{1}{j}\right)\frac{4d(x,y)}{\Delta}$

$d_T(x,y)$ if cut level Δ is 2Δ.
$\rightarrow E[d_T(x,y)] = \sum_{\Delta = \frac{D}{2^i}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y)$

The pipes are distinct!

$$E(d_T(x,y)] = \sum_{\Delta = D/2^i} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$$

The pipes are distinct!

$$E(d_T(x,y)] = \sum_{\Delta=D/2^i} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

The pipes are distinct!

$E(d_T(x,y)] = \sum_{\Delta = D/2^i} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

"Listen Stash, the pipes are distinct!!"

The pipes are distinct!

$E(d_T(x,y)] = \sum_{\Delta=D/2^i} \sum_{j\in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

“Listen Stash, the pipes are distinct!!”

Uh.. well X_Δ is distinct from $X_{\Delta/4}$.

The pipes are distinct!

$E(d_T(x,y)] = \sum_{\Delta = D/2^i} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

"Listen Stash, the pipes are distinct!!"

Uh.. well X_Δ is distinct from $X_{\Delta/4}$.

$E(d_T(x,y)] = \sum_{\Delta = \frac{D}{4^i}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y) + \sum_{\Delta = \frac{D}{((2)4^i)}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y)$

The pipes are distinct!

$E(d_T(x,y)] = \sum_{\Delta = D/2^i} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

"Listen Stash, the pipes are distinct!!"

Uh.. well X_Δ is distinct from $X_{\Delta/4}$.

$$E(d_T(x,y)] = \sum_{\Delta = \frac{D}{4^i}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y) + \sum_{\Delta = \frac{D}{((2)4^i)}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y)$$

$$\leq 2 \sum_j \left(\frac{1}{j}\right) 4d(x,y)$$

The pipes are distinct!

$E(d_T(x,y)] = \sum_{\Delta=D/2^i} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

"Listen Stash, the pipes are distinct!!"

Uh.. well X_Δ is distinct from $X_{\Delta/4}$.

$E(d_T(x,y)] = \sum_{\Delta=\frac{D}{4^i}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y) + \sum_{\Delta=\frac{D}{((2)4^i)}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y)$

$\leq 2\sum_j \left(\frac{1}{j}\right) 4d(x,y)$

$\leq (16 \ln n)(d(x,y))$.

The pipes are distinct!

$E(d_T(x,y)] = \sum_{\Delta=D/2^i} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

"Listen Stash, the pipes are distinct!!"

Uh.. well X_Δ is distinct from $X_{\Delta/4}$.

$E(d_T(x,y)] = \sum_{\Delta=\frac{D}{4^i}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y) + \sum_{\Delta=\frac{D}{((2)4^i)}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y)$
$\leq 2\sum_j \left(\frac{1}{j}\right) 4d(x,y)$
$\leq (16 \ln n)(d(x,y))$.

Claim: $E[d_T(x,y)] = O(logn)d(x,y)$

The pipes are distinct!

$E(d_T(x,y)] = \sum_{\Delta=D/2^i} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

"Listen Stash, the pipes are distinct!!"

Uh.. well X_Δ is distinct from $X_{\Delta/4}$.

$$E(d_T(x,y)] = \sum_{\Delta=\frac{D}{4^i}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y) + \sum_{\Delta=\frac{D}{((2)4^i)}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y)$$
$$\leq 2\sum_j \left(\frac{1}{j}\right) 4d(x,y)$$
$$\leq (16 \ln n)(d(x,y)).$$

Claim: $E[d_T(x,y)] = O(logn)d(x,y)$

Expected stretch is $O(\log n)$.

The pipes are distinct!

$E(d_T(x,y)] = \sum_{\Delta = D/2^i} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

"Listen Stash, the pipes are distinct!!"

Uh.. well X_Δ is distinct from $X_{\Delta/4}$.

$$E(d_T(x,y)] = \sum_{\Delta = \frac{D}{4^i}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y) + \sum_{\Delta = \frac{D}{((2)4^i)}} \sum_{j \in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y)$$
$$\leq 2\sum_j \left(\frac{1}{j}\right) 4d(x,y)$$
$$\leq (16 \ln n)(d(x,y)).$$

Claim: $E[d_T(x,y)] = O(logn)d(x,y)$

Expected stretch is $O(\log n)$.

We gave an algorithm that produces a distribution of trees.

The pipes are distinct!

$E(d_T(x,y)] = \sum_{\Delta=D/2^i} \sum_{j\in X_\Delta} \left(\frac{1}{j}\right) 2d(x,y)$

Recall X_Δ has nodes with $d(x,j) \in [\Delta/4, \Delta/2]$

"Listen Stash, the pipes are distinct!!"

Uh.. well X_Δ is distinct from $X_{\Delta/4}$.

$$E(d_T(x,y)] = \sum_{\Delta=\frac{D}{4^i}} \sum_{j\in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y) + \sum_{\Delta=\frac{D}{((2)4^i)}} \sum_{j\in X_\Delta} \left(\frac{1}{j}\right) 8d(x,y)$$
$$\leq 2\sum_j \left(\frac{1}{j}\right) 4d(x,y)$$
$$\leq (16\ln n)(d(x,y)).$$

Claim: $E[d_T(x,y)] = O(logn)d(x,y)$

Expected stretch is $O(\log n)$.

We gave an algorithm that produces a distribution of trees.

The expected stretch of any pair is $O(\log n)$.

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?
Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.

(Disguised a bit.)

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.
(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.

(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Given solution to toll problem, find cut?

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.
(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Given solution to toll problem, find cut?

Top level cuts each edge with prob. $O(\log n)/D$, D is diameter.

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.
(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Given solution to toll problem, find cut?

Top level cuts each edge with prob. $O(\log n)/D$, D is diameter. D is at least average distance: $\sum_{i,j} d(i,j)/n^2$.

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.
(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Given solution to toll problem, find cut?

Top level cuts each edge with prob. $O(\log n)/D$, D is diameter. D is at least average distance: $\sum_{i,j} d(i,j)/n^2$.

If cut is balanced

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.
(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Given solution to toll problem, find cut?

Top level cuts each edge with prob. $O(\log n)/D$, D is diameter. D is at least average distance: $\sum_{i,j} d(i,j)/n^2$.

If cut is balanced $|S||\overline{S}|$ is $\Theta(n^2)$

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?
Cheeger: approximately find small expansion cut. (Quadratic approximation.)
Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.
(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Given solution to toll problem, find cut?

Top level cuts each edge with prob. $O(\log n)/D$, D is diameter. D is at least average distance: $\sum_{i,j} d(i,j)/n^2$.

If cut is balanced $|S||\overline{S}|$ is $\Theta(n^2)$
and sparsity is $\frac{O(\log n)/D}{cn^2} = \frac{O(\log n)}{\sum_{i,j} d(i,j)}$.

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?

Cheeger: approximately find small expansion cut. (Quadratic approximation.)

Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.
(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Given solution to toll problem, find cut?

Top level cuts each edge with prob. $O(\log n)/D$, D is diameter. D is at least average distance: $\sum_{i,j} d(i,j)/n^2$.

If cut is balanced $|S||\overline{S}|$ is $\Theta(n^2)$
and sparsity is $\frac{O(\log n)/D}{cn^2} = \frac{O(\log n)}{\sum_{i,j} d(i,j)}$.

$\rightarrow$ find $O(\log n)$ times optimal sparse cut.

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?
Cheeger: approximately find small expansion cut. (Quadratic approximation.)
Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.
(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Given solution to toll problem, find cut?

Top level cuts each edge with prob. $O(\log n)/D$, D is diameter. D is at least average distance: $\sum_{i,j} d(i,j)/n^2$.

If cut is balanced $|S||\overline{S}|$ is $\Theta(n^2)$
and sparsity is $\frac{O(\log n)/D}{cn^2} = \frac{O(\log n)}{\sum_{i,j} d(i,j)}$.

$\rightarrow$ find $O(\log n)$ times optimal sparse cut.

If not...

Alternative to Cheeger for expansion.

Graph G, sparsity of cut $\frac{E(S,\overline{S})}{|S|\overline{S}}$,

Find smallest sparsity cut?
Cheeger: approximately find small expansion cut. (Quadratic approximation.)
Recall: Expansion estimates sparsity within factor of two.

Toll problem: assign tolls to max. average toll bet. all pairs of vertices.

Exam: Sparsity of graph is lower bounded by function of toll problem.
(Disguised a bit.)

Lemma: $\mathscr{S} \geq \frac{1}{\sum_{i,j} d(i,j)}$.

Given solution to toll problem, find cut?

Top level cuts each edge with prob. $O(\log n)/D$, D is diameter. D is at least average distance: $\sum_{i,j} d(i,j)/n^2$.

If cut is balanced $|S||\overline{S}|$ is $\Theta(n^2)$
and sparsity is $\frac{O(\log n)/D}{cn^2} = \frac{O(\log n)}{\sum_{i,j} d(i,j)}$.

$\rightarrow$ find $O(\log n)$ times optimal sparse cut.

If not...a bit more work...

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Find an labeling of vertices, $\ell : V \to X$ that minimizes

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Find an labeling of vertices, $\ell : V \to X$ that minimizes

$\sum_{e=(u,v)} c(e) d(l(u), l(v)) + \sum_v c(v, l(v))$

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Find an labeling of vertices, $\ell : V \to X$ that minimizes

$\sum_{e=(u,v)} c(e) d(l(u), l(v)) + \sum_v c(v, l(v))$

Idea: find HST for metric (X, d).

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Find an labeling of vertices, $\ell : V \rightarrow X$ that minimizes

$\sum_{e=(u,v)} c(e) d(l(u), l(v)) + \sum_v c(v, l(v))$

Idea: find HST for metric (X, d).

Solve the problem on a hierarchically well separated tree metric.

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Find an labeling of vertices, $\ell : V \rightarrow X$ that minimizes

$\sum_{e=(u,v)} c(e) d(l(u), l(v)) + \sum_v c(v, l(v))$

Idea: find HST for metric (X, d).

Solve the problem on a hierarchically well separated tree metric.

Kleinberg-Tardos: constant factor on uniform metric.

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Find an labeling of vertices, $\ell : V \to X$ that minimizes

$\sum_{e=(u,v)} c(e) d(l(u), l(v)) + \sum_v c(v, l(v))$

Idea: find HST for metric (X, d).

Solve the problem on a hierarchically well separated tree metric.

Kleinberg-Tardos: constant factor on uniform metric.

Hierarchically well separated tree,

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Find an labeling of vertices, $\ell : V \to X$ that minimizes

$\sum_{e=(u,v)} c(e) d(l(u), l(v)) + \sum_v c(v, l(v))$

Idea: find HST for metric (X, d).

Solve the problem on a hierarchically well separated tree metric.

Kleinberg-Tardos: constant factor on uniform metric.

Hierarchically well separated tree, “geometric”,

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Find an labeling of vertices, $\ell : V \rightarrow X$ that minimizes

$\sum_{e=(u,v)} c(e) d(l(u), l(v)) + \sum_v c(v, l(v))$

Idea: find HST for metric (X, d).

Solve the problem on a hierarchically well separated tree metric.

Kleinberg-Tardos: constant factor on uniform metric.

Hierarchically well separated tree, “geometric”, constant factor.

Metric Labelling

Input: graph $G = (V, E)$ with edge weights, $w(\cdot)$, metric labels (X, d), and costs for mapping vertices to labels $c : V \times X$.

Find an labeling of vertices, $\ell : V \rightarrow X$ that minimizes

$\sum_{e=(u,v)} c(e) d(l(u), l(v)) + \sum_{v} c(v, l(v))$

Idea: find HST for metric (X, d).

Solve the problem on a hierarchically well separated tree metric.

Kleinberg-Tardos: constant factor on uniform metric.

Hierarchically well separated tree, "geometric", constant factor.

$\rightarrow O(\log n)$ approximation.

See you ...

Tuesday.