Steiner Point Removal with Distortion $O(\log k)$

Arnold Filtser

Ben-Gurion University

April 26, 2018
Graph Minor

$H$ is a **minor** of $G = (V, E)$ if $H$ can be **formed** from $G$ by:

- Deleting edges.
- Deleting vertices.
- Contracting edges.
Graph Minor

$H$ is a **minor** of $G = (V, E)$ if $H$ can be **formed** from $G$ by:

- Deleting edges.

![Diagram](image-url)
Graph Minor

$H$ is a **minor** of $G = (V, E)$ if $H$ can be **formed** from $G$ by:

- Deleting edges.

- Deleting vertices.

---

[Diagram showing deletion of edges and vertices]
Graph Minor

H is a \textbf{minor} of G = (V, E) if H can be \textbf{formed} from G by:

- Deleting edges.

- Deleting vertices.

- Contracting edges.
Steiner Point removal problem

\[ G = (V, E, w) \] - a \textbf{weighted} graph.

\[ K \subseteq V \] - a \textbf{terminal} set of size \( k \).
Steiner Point removal problem

\[ G = (V, E, w) \] - a weighted graph.
\[ K \subseteq V \] - a terminal set of size \( k \).

Construct a new graph \( M = (K, E', w_M) \) such that:

\[ \forall t, t' \in K, d_G(t, t') \leq d_M(t, t') \leq \alpha \cdot d_G(t, t') \]

\( M \) is a graph minor of \( G \).
Steiner Point removal problem

$G = (V, E, w)$ - a weighted graph.

$K \subseteq V$ - a terminal set of size $k$.

Construct a new graph $M = (K, E', w_M)$ such that:

- $M$ has small distortion:

  $$\forall t, t' \in K, \quad d_G(t, t') \leq d_M(t, t') \leq \alpha \cdot d_G(t, t') .$$
Steiner Point removal problem

$G = (V, E, w)$ - a weighted graph.
$K \subseteq V$ - a terminal set of size $k$.

Construct a new graph $M = (K, E', w_M)$ such that:

- $M$ has small distortion:
  $\forall t, t' \in K, \ d_G(t, t') \leq d_M(t, t') \leq \alpha \cdot d_G(t, t')$.

- $M$ is a graph minor of $G$. 
Steiner Point removal problem

$G = (V, E, w)$ - a **weighted** graph.

$K \subseteq V$ - a **terminal** set of size $k$.

Construct a new graph $M = (K, E', w_M)$ such that:

- $M$ has small **distortion**:

  $\forall t, t' \in K, \ d_G(t, t') \leq d_M(t, t') \leq \alpha \cdot d_G(t, t')$.

- $M$ is a graph **minor** of $G$.

The distortion is:

$$\frac{d_M(t, t')}{d_G(t, t')} = \frac{4}{2} = 2$$
Terminal Partitions and Induced Minor

Partition \( \{ V_1, \ldots, V_k \} \) of \( V \) is called a **terminal partition** if for all \( i \),

- \( t_i \in V_i \).
- \( V_i \) is **connected**.

![Diagram](https://example.com/diagram.png)
Partition \{ V_1, \ldots, V_k \} of \( V \) is called a terminal partition if for all \( i \),

- \( t_i \in V_i \).
- \( V_i \) is connected.

Given a terminal partition \( P = \{ V_1, \ldots, V_k \} \), the induced minor \( M \) is obtained by contracting all the internal edges in each \( V_i \).
Terminal Partitions and Induced Minor

Partition \( \{ V_1, \ldots, V_k \} \) of \( V \) is called a \textbf{terminal partition} if for all \( i \),

\begin{itemize}
  \item \( t_i \in V_i \).
  \item \( V_i \) is \textbf{connected}.
\end{itemize}

Given a terminal partition \( P = \{ V_1, \ldots, V_k \} \), the \textbf{induced minor} \( M \) is obtained by \textbf{contracting} all the internal edges in each \( V_i \).

![Diagram of terminal partitions and induced minor]
Terminal Partitions and Induced Minor

Partition \( \{ V_1, \ldots, V_k \} \) of \( V \) is called a terminal partition if for all \( i \),

- \( t_i \in V_i \).
- \( V_i \) is connected.

Given a terminal partition \( P = \{ V_1, \ldots, V_k \} \), the induced minor \( M \) is obtained by contracting all the internal edges in each \( V_i \).

The weight of \( \{ t, t' \} \) (if exist) is simply \( d_G(t, t') \).
Terminal Partitions and Induced Minor

Partition \( \{ V_1, \ldots, V_k \} \) of \( V \) is called a **terminal partition** if for all \( i \),

- \( t_i \in V_i \).
- \( V_i \) is **connected**.

Given a terminal partition \( P = \{ V_1, \ldots, V_k \} \), the **induced minor** \( M \) is obtained by **contracting** all the internal edges in each \( V_i \).

The **weight** of \( \{ t, t' \} \) (if exist) is simply \( d_G(t, t') \).
Terminal Partitions and Induced Minor

Partition \( \{ V_1, \ldots, V_k \} \) of \( V \) is called a **terminal partition** if for all \( i \),

- \( t_i \in V_i \).
- \( V_i \) is connected.

Given a terminal partition \( P = \{ V_1, \ldots, V_k \} \), the **induced minor** \( M \) is obtained by **contracting** all the internal edges in each \( V_i \).

The **weight** of \( \{ t, t' \} \) (if exist) is simply \( d_G(t, t') \).

The distortion is:

\[ \frac{d_M(t_1, t_3)}{d_G(t_1, t_3)} = \frac{12}{4} = 3. \]
Induced Minor by Voronoi Cells

Natural candidate:
Let $V_j$ be the **Voronoi cell** of $t_j$ (breaking ties arbitrarily).

$$V_j = \{ v \in V | \forall i \neq j \quad d_G(t_j, v) \leq d_G(t_i, v) \}$$
Natural candidate:
Let $V_j$ be the **Voronoi cell** of $t_j$ (breaking ties arbitrarily).

$$V_j = \{ v \in V \mid \forall i \neq j \quad d_G(t_j, v) \leq d_G(t_i, v) \}$$
Induced Minor by Voronoi Cells

Natural candidate:
Let $V_j$ be the Voronoi cell of $t_j$ (breaking ties arbitrarily).

$$V_j = \{ v \in V \mid \forall i \neq j \quad d_G(t_j, v) \leq d_G(t_i, v) \}$$
Induced Minor by Voronoi Cells

Natural candidate:
Let $V_j$ be the **Voronoi cell** of $t_j$ (breaking ties arbitrarily).

$$V_j = \{ v \in V \mid \forall i \neq j \quad d_G(t_j, v) \leq d_G(t_i, v) \}$$
Induced Minor by Voronoi Cells

Natural candidate:
Let $V_j$ be the Voronoi cell of $t_j$ (breaking ties arbitrarily).

\[ V_j = \{ v \in V \mid \forall i \neq j \quad d_G(t_j, v) \leq d_G(t_i, v) \} \]
Gupta (2001) showed upper bound of 8 for trees.
1. Gupta (2001) showed **upper bound** of 8 for trees.
2. Chan, Xia, Konjevod, and Richa (2006) showed: **lower bound** of 8 for trees.
History

1. Gupta (2001) showed **upper bound** of 8 for trees.

2. Chan, Xia, Konjevod, and Richa (2006) showed: **lower bound** of 8 for trees.

   Best known lower bound for general graphs!
History

1. Gupta (2001) showed upper bound of 8 for trees.
2. Chan, Xia, Konjevod, and Richa (2006) showed: lower bound of 8 for trees.
   Best known lower bound for general graphs!
3. Basu and Gupta (2008) showed upper bound of $O(1)$ for outerplanar graphs.
History

1. Gupta (2001) showed **upper bound** of 8 for trees.
2. Chan, Xia, Konjevod, and Richa (2006) showed: **lower bound** of 8 for trees.

   Best known lower bound for general graphs!

3. Basu and Gupta (2008) showed **upper bound** of $O(1)$ for outerplanar graphs.
4. Kamma, Krauthgamer and Nguyen (2014) showed **upper bound** of $O(\log^6 k)$ for general graphs.
History

1. Gupta (2001) showed upper bound of 8 for trees.
2. Chan, Xia, Konjevod, and Richa (2006) showed: lower bound of 8 for trees.

Best known lower bound for general graphs!

3. Basu and Gupta (2008) showed upper bound of $O(1)$ for outerplanar graphs.
4. Kamma, Krauthgamer and Nguyen (2014) showed upper bound of $O(\log^6 k)$ for general graphs.

Using the Ball growing algorithm.
Gupta (2001) showed **upper bound** of $8$ for trees.

Chan, Xia, Konjevod, and Richa (2006) showed:

**lower bound** of $8$ for trees.

Best known lower bound for general graphs!

Basu and Gupta (2008) showed

**upper bound** of $O(1)$ for **outerplanar** graphs.

Kamma, Krauthgamer and Nguyen (2014) showed

**upper bound** of $O(\log^6 k)$ for general graphs.

Using the **Ball growing algorithm**.

Kamma, Krauthgamer and Nguyen (2015)

improved analysis to $O(\log^5 k)$ (same alg).
History

1. Gupta (2001) showed **upper bound** of 8 for trees.
2. Chan, Xia, Konjevod, and Richa (2006) showed:
   - **lower bound** of 8 for trees.
   - Best known lower bound for general graphs!
3. Basu and Gupta (2008) showed
   - **upper bound** of $O(1)$ for **outerplanar** graphs.
4. Kamma, Krauthgamer and Nguyen (2014) showed
   - **upper bound** of $O(\log^6 k)$ for general graphs.
   - Using the **Ball growing algorithm**.
5. Kamma, Krauthgamer and Nguyen (2015)
   - improved analysis to $O(\log^5 k)$ (same alg).
6. Cheung (2018) improved analysis to $O(\log^2 k)$ (same alg).
Results

Obtain improved the analysis of the Ball Growing algorithm to $O(\log k)$. 

Introduce a new algorithm: The Noisy Voronoi algorithm. Also induce distortion of $O(\log k)$. Simpler analysis. Can be implemented in almost linear time! ($O(m \log n)$).
Obtain improved the analysis of the Ball Growing algorithm to $O(\log k)$.

(Appeared in SODA 18)
Results

Obtain improved the analysis of the Ball Growing algorithm to $O(\log k)$.

(Appeared in SODA 18)

Introduce a new algorithm: The Noisy Voronoi algorithm.
- Also induce distortion of $O(\log k)$.
Obtain improved the analysis of the Ball Growing algorithm to $O(\log k)$.

(Appeared in SODA 18)

Introduce a new algorithm: The **Noisy Voronoi** algorithm.
- Also induce distortion of $O(\log k)$.
- **Simpler** analysis.
Obtain improved the analysis of the Ball Growing algorithm to $O(\log k)$.

(Appeared in SODA 18)

Introduce a new algorithm: The **Noisy Voronoi** algorithm.

- Also induce distortion of $O(\log k)$.
- **Simpler** analysis.
- Can be implemented in almost linear time! ($O(m \log n)$).
Results

Obtain improved the analysis of the Ball Growing algorithm to $O(\log k)$.

(Appeared in SODA 18)

Introduce a new algorithm: The Noisy Voronoi algorithm.
- Also induce distortion of $O(\log k)$.
- **Simpler** analysis.
- Can be implemented in almost **linear time!** ($O(m \log n)$).
Arbitrary order of the terminals
$R = 1.53$

Distribution to be specified later

$R = O(1) \text{ w.h.p}$
$R = 1.53$
\( R = 1.53 \)
$R = 1.53$
\[ D(v) = \min_{t \in K} d(v, t) = 2 \]
$R = 1.53$

$D(v) = 2$

$d(v, t_1) = 3$

$D(v) = \min_{t \in K} d(v, t) = 2$
\( D(v) = \min_{t \in K} d(v, t) = 2 \)

\( d(v, t_1) \leq R \cdot D(v) \)

\( d(v, t_1) = 3 \)

\( R = 1.53 \)
$R = 1.53$
$R = 1.53$
$R = 1.53$

$D(v) = 2$
$R = 1.53$

$D(v) = 2$

$d(v, t_1) = 4$
$R = 1.53$

$D(v) = 2$

$d(v, t_1) = 4$

$d(v, t_1) > R \cdot D(v)$
$R = 1.53$

$D(v) = 3$
\[ R = 1.53 \]
\[ D(v) = 3 \]
\[ d(v, t_1) = 5 \]
$$R = 1.53$$

$$D(v) = 3$$

$$d(v, t_1) > R \cdot D(v)$$

$$d(v, t_1) = 5$$
\[ R = 1.53 \]

\[ D(v) = 4 \]
\[ R = 1.53 \]
\[ D(v) = 4 \]
\[ d(v, t_1) = 6 \]
\[ d(v, t_1) \leq R \cdot D(v) \]

\[ R = 1.53 \]

\[ D(v) = 4 \]

\[ d(v, t_1) = 6 \]
$R = 1.53$
$R = 1.53$
$V_1$
$R = 1.3$

$R = 1.53$

$V_1$
$R = 1.3$

$R = 1.53$

$V_1$
$R = 1.3$

$R = 1.53$

$V_1$

$V_2$
$R = 1.3$

$R = 1.53$

$V_1$

$R = 1.6$

$V_2$
\( R = 1.53 \)

\( V_1 \)

\( R = 1.3 \)

\( V_2 \)

\( R = 1.6 \)

\( V_3 \)

\( R = 1.1 \)
$R = 1.3$

$R = 1.6$

$R = 1.1$

$R = 1.53$

$V_1$

$V_2$

$V_3$

$V_4$
$R = 1.53$

$V_1$

$R = 1.3$

$V_2$

$R = 1.6$

$V_3$

$R = 1.1$

$V_4$

$R = 2$
Noisy Voronoi

Set $\delta = \frac{1}{20} \ln k$ and $p = \frac{1}{5}$.

Set $R_j \leftarrow (1 + \delta)^{g_j}$, where $g_j \sim \text{Geo}(p)$. 

Lemma

The Noisy Voronoi algorithm creates a terminal partition.
Noisy Voronoi

Set $\delta = \frac{1}{20} \ln k$ and $p = \frac{1}{5}$.

Set $R_j \leftarrow (1 + \delta)^{g_j}$, where $g_j \sim \text{Geo}(p)$.

Note that

$$g_j = O(\log k) \text{ (w.h.p)} \quad \Rightarrow \quad R_j = O(1).$$
Noisy Voronoi

Set $\delta = \frac{1}{20 \ln k}$ and $p = \frac{1}{5}$.

Set $R_j \leftarrow (1 + \delta)^{g_j}$, where $g_j \sim \text{Geo}(p)$.

Note that

$g_j = O(\log k) \ (\text{w.h.p}) \quad \Rightarrow \quad R_j = O(1)$.

If $v$ joins $V_j$, the cluster of $t_j$, then

$d(v, t_j) \leq R_j \cdot D(v) = O(D(v))$. 
Noisy Voronoi

Set $\delta = \frac{1}{20} \ln k$ and $p = \frac{1}{5}$.

Set $R_j \leftarrow (1 + \delta)^{g_j}$, where $g_j \sim \text{Geo}(p)$.

Note that

$$g_j = O(\log k) \text{ (w.h.p)} \implies R_j = O(1).$$

If $v$ joins $V_j$, the cluster of $t_j$, then

$$d(v, t_j) \leq R_j \cdot D(v) = O(D(v)).$$

Lemma

The Noisy Voronoi algorithm creates a terminal partition.
The Seed of Evil (distortion)

\[ t, t' \in K, \ P_{t,t'} \ is \ a \ shortest \ path \ in \ G. \]
$t, t' \in K$, $P_{t,t'}$ is a shortest path in $G$. 

$$P_{t,t'}$$
$t, t' \in K$, $P_{t,t'}$ is a shortest path in $G$.
$v_{\ell_i}$ is arbitrary vertex on $P_{t,t'}$ covered by $t_{\ell_i}$.
The Seed of Evil (distortion)

t, t' ∈ K, \( P_{t,t'} \) is a shortest path in \( G \).
v_{\ell_i} is arbitrary vertex on \( P_{t,t'} \) covered by \( t_{\ell_i} \).
The Seed of Evil (distortion)

$t, t' \in K$, $P_{t,t'}$ is a shortest path in $G$.

$v_{\ell_i}$ is arbitrary vertex on $P_{t,t'}$ covered by $t_{\ell_i}$. 
\begin{align*}
d_M(t, t') & \leq d_G(t, t') + 2 \sum_i d_G(t_{\ell_i}, v_{\ell_i}) 
\end{align*}
The Seed of Evil (distortion)

\[ d_M(t, t') \leq d_G(t, t') + 2 \sum_i d_G(t_{\ell_i}, v_{\ell_i}) \]

Analyze \( \sum_i d_G(t_{\ell_i}, v_{\ell_i}) \)!
Analyzing $\sum_i d_G(t_i, v_i)$ directly will be tricky, as $d_G(t_i, v_i)$ depends on $V_1, \ldots, V_{i-1}$. 
Analyzing $\sum_i d_G(t_i, v_i)$ directly will be tricky, as $d_G(t_i, v_i)$ depends on $V_1, \ldots, V_{i-1}$.

We will partition $P_{t,t'}$ into intervals, and charge the interval starting the detour instead of the terminal!
Analyzing $\sum_i d_G(t_i, v_i)$ directly will be tricky, as $d_G(t_i, v_i)$ depends on $V_1, \ldots, V_{i-1}$. We will partition $P_{t,t'}$ into intervals, and charge the interval starting the detour instead of the terminal!
Partition of $P_{t,t'}$ to Intervals

$Q$ is a interval of $P_{t,t'}$.

\[
L(Q) = d_G(v_a, v_b)
\]

Interval length
Partition of $P_{t,t'}$ to Intervals

$Q$ is a interval of $P_{t,t'}$.

$$D(Q) = \Theta(\log k) \cdot L(Q)$$

$$L(Q) = d_G(v_a, v_b)$$

Partition $P_{t,t}$ into $Q$, s.t. for each $Q \in Q$

$$L(Q) = \Theta\left(\frac{1}{\log k}\right) \cdot D(Q)$$
Partition of $P_{t,t'}$ to Intervals

$Q$ is a interval of $P_{t,t'}$.

\[ L(Q) = d_G(v_a, v_b) \]

\[ D(Q) = \Theta(\log k) \cdot L(Q) \]

Interval length

Partition $P_{t,t}$ into $Q$, s.t. for each $Q \in \mathcal{Q}$

\[ L(Q) = \Theta\left(\frac{1}{\log k}\right) \cdot D(Q) \]
Partition of $P_{t,t'}$ to Intervals

$Q$ is a interval of $P_{t,t'}$.

$D(Q) = \Theta(\log k) \cdot L(Q)$

$\forall j \in \{1, \ldots, Kn\}$

$L(Q) = d_G(v_a, v_b)$

Interval length

Partition $P_{t,t}$ into $Q$, s.t. for each $Q \in Q$

$L(Q) = \Theta\left(\frac{1}{\log k}\right) \cdot D(Q)$

Once $t_j$ covered some $v_j \in Q$, w.p $1 - p$ it covers all of $Q$. 
Active vertices

At the beginning all vertices are active.
Active vertices

At the beginning all vertices are active.

Terminal $t_j$ grows cluster $V_j$. 
Active vertices

At the beginning all vertices are active.

Terminal $t_j$ grows cluster $V_j$. $a_j$ (resp. $b_j$) is the leftmost (resp. rightmost) active covered vertex.
Active vertices

At the beginning all vertices are active.

Terminal \( t_j \) grows cluster \( V_j \).
\( a_j \) (resp. \( b_j \)) is the leftmost (resp. rightmost) active covered vertex.
\( D_j = \{a_j, \ldots, b_j\} \subseteq P_{t,t'} \) is called a detour.
Active vertices

At the beginning all vertices are active.

Terminal $t_j$ grows cluster $V_j$. $a_j$ (resp. $b_j$) is the leftmost (resp. rightmost) active covered vertex. $\mathcal{D}_j = \{a_j, \ldots, b_j\} \subseteq P_{t, t'}$ is called a detour.

All the vertices in $\mathcal{D}_j$ become inactive.
Active vertices

At the beginning all vertices are active.

Terminal $t_j$ grows cluster $V_j$. $a_j$ (resp. $b_j$) is the leftmost (resp. rightmost) active covered vertex. $\mathcal{D}_j = \{a_j, \ldots, b_j\} \subseteq P_{t,t'}$ is called a detour. All the vertices in $\mathcal{D}_j$ become inactive.
Active vertices

At the beginning all vertices are \textbf{active}.

Terminal $t_j$ grows cluster $V_j$. $a_j$ (resp. $b_j$) is the leftmost (resp. rightmost) \textbf{active} covered vertex. $\mathcal{D}_j = \{a_j, \ldots, b_j\} \subseteq P_{t,t'}$ is called a \textbf{detour}. All the vertices in $\mathcal{D}_j$ become \textbf{inactive}.
Charges

\[ R_j = (1 + \delta)^{g_1} \]

Detour \( D_j \) will be **charged** upon a single interval.
Charges

\[ R_j = (1 + \delta)^{g_2} \]

Detour \( D_j \) will be \textbf{charged} upon a single interval.
Charges

$$R_j = (1 + \delta)^g$$

Detour $D_j$ will be **charged** upon a single interval. $v_j$ is the **first active** covered vertex by $t_j$ in $P_{t,t'}$. 
Charges

\[ R_j = (1 + \delta)^{g_4} \]

Detour \( D_j \) will be **charged** upon a single interval. \( v_j \) is the “**first active**” covered vertex by \( t_j \) in \( P_{t,t'} \).
Charges

$R_j = (1 + \delta)g_5$

Detour $D_j$ will be charged upon a single interval. $v_j$ is the "first active" covered vertex by $t_j$ in $P_{t,t'}$. 


Charges

\[ R_j = (1 + \delta)g5 \]

Detour \( D_j \) will be **charged** upon a single interval. 

\( v_j \) is the "**first active**" covered vertex by \( t_j \) in \( P_{t,t'} \).

\( Q_j \in Q \) (\( v_j \in Q_j \)) is charged upon \( D_j \).
Charges

\[ R_j = (1 + \delta)^{g_5} \]

Detour \( \mathcal{D}_j \) will be **charged** upon a single interval.

\( v_j \) is the “**first active**” covered vertex by \( t_j \) in \( P_{t,t'} \).

\( Q_j \in \mathcal{Q} \) (\( v_j \in Q_j \)) is charged upon \( \mathcal{D}_j \).

\( X_Q \) is the **current** number of detours the interval \( Q \) is **charged** for.
Detour $D_j$ will be **charged** upon a single interval.

$v_j$ is the “**first active**” covered vertex by $t_j$ in $P_{t,t'}$.

$Q_j \in Q$ ($v_j \in Q_j$) is charged upon $D_j$.

$X_Q$ is the **current** number of detours the interval $Q$ is **charged** for.
Charges

$R_j = (1 + \delta)^{g_5}$

Detour $D_j$ will be charged upon a single interval. $v_j$ is the “first active” covered vertex by $t_j$ in $P_{t,t'}$. $Q_j \in Q$ ($v_j \in Q_j$) is charged upon $D_j$. $X_Q$ is the current number of detours the interval $Q$ is charged for. Every detour $D_{j'}$ which is contained in $D_j$ erased, and its charge re-funded!
Charges

\[ R_j = (1 + \delta)g^5 \]

\[ X_{Q_j} \] increases by \textbf{at most 1}. 
Charges

\[ R_j = (1 + \delta)^{g_5} \]

\[ X_{Q_2} = 1 \quad X_{Q_4} = 1 \]

\[ X_{Q_1} = 0 \quad X_{Q_3} = 0 \]

\[ v_j \]

\[ Q^1 \quad Q^2 = Q_j \quad Q^3 \quad Q^4 \]

\[ X_{Q_j} \text{ increases by at most 1.} \]

For every \( Q \neq Q_j \), \( X_Q \) can only decrease.
Slices: “The Potential to be Charged”

Within interval $Q \in \mathcal{Q}$, maximal sub-interval of active vertices is called a slice.
Slices: “The Potential to be Charged”

Within interval $Q \in Q$, maximal sub-interval of active vertices is called a slice.

We denote by $\#S(Q)$ the current number of slices in $Q$. 

At the start, $\#S(Q) = 1$. At the end, $\#S(Q) = 0$. 
Slices: “The Potential to be Charged”

Within interval $Q \in \mathcal{Q}$, maximal sub-interval of active vertices is called a slice.

We denote by $\#S(Q)$ the current number of slices in $Q$.

At the start, $\#S(Q) = 1$. At the end, $\#S(Q) = 0$. 
Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q_j$. 

Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q_j$.

$\#S(Q_j)$ can increase by 1.
Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q_j$.

$\#S(Q_j)$ can **increase** by 1.

$\#S(Q_j)$ can **decrease**.

$\#S(Q_j)$ can **stay unchanged**.
Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q_j$.

$\#S(Q_j)$ can **increase** by 1.

$\#S(Q_j)$ can **decrease**.

$\#S(Q_j)$ can stay **unchanged**.

Arnold Filtser
Steiner Point Removal
April 26, 2018
Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q_j$.

In any case, $\#S(Q_j)$ can increase by at most $1!$
Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q_j$.

In any case, $\#S(Q_j)$ can increase by at most 1!

If $\#S(Q_j)$ is decreased, we call it a success.
Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q_j$.

In any case, $\#S(Q_j)$ can increase by at most 1!

If $\#S(Q_j)$ is decreased, we call it a success.

Otherwise, we call it a failure.
Change in Number of Slices

Let \( S_j \subseteq Q_j \) be the slice containing \( v_j \). Consider \( Q \neq Q_j \).
Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q \neq Q_j$.

# $S(Q)$ can decrease.

In any case, # $S(Q)$ cannot increase!
Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q \neq Q_j$.

$\#S(Q)$ can **decrease**.

$\#S(Q)$ can stay **unchanged**.
Change in Number of Slices

Let $S_j \subseteq Q_j$ be the slice containing $v_j$. Consider $Q \neq Q_j$.

$\#S(Q)$ can **decrease**.

In any case, $\#S(Q)$ cannot increase!
Lemma (Success probability)

Assuming at least one active vertex joins $V_j$,

the probability of success is at least $1 - p$. 
Lemma (Success probability)

Assuming at least one active vertex joins $V_j$, the probability of success is at least $1 - p$. 

$$t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j)$$
Lemma (Success probability)

Assuming at least one active vertex joins $V_j$, the probability of success is at least $1 - p$.

$$t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j)$$

$$v_j \in V_j \implies R_j \geq d(v_j, t_j)/D(v_j)$$
Lemma (Success probability)

Assuming at least one active vertex joins $V_j$, the probability of success is at least $1 - p$.

$t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j)$

$v_j \in V_j \Rightarrow R_j \geq d(v_j, t_j) / D(v_j)$. For all $z \in S_j$, 
Lemma (Success probability)

Assuming at least one active vertex joins $V_j$, the probability of success is at least $1 - p$.

\[ t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j) \]

For all $z \in S_j$,

\[ \frac{d(z, t_j)}{D(z)} \leq \frac{d(v_j, t_j) + L(Q_j)}{D(v_j) - L(Q_j)} \leq \frac{d(v_j, t_j)}{D(v_j)} \left(1 + \frac{O(1)}{\log k}\right) \]

$v_j \in V_j \Rightarrow R_j \geq d(v_j, t_j) / D(v_j)$. For all $z \in S_j$, \[ \frac{d(z, t_j)}{D(z)} \leq \frac{d(v_j, t_j) + L(Q_j)}{D(v_j) - L(Q_j)} \leq \frac{d(v_j, t_j)}{D(v_j)} \left(1 + \frac{O(1)}{\log k}\right) \]
Lemma (Success probability)

Assuming at least one active vertex joins $V_j$, the probability of success is at least $1 - p$. 

\[ t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j) \]

\[ v_j \in V_j \quad \Rightarrow \quad R_j \geq d(v_j, t_j)/D(v_j). \] 

For all $z \in S_j$, 

\[ \frac{d(z,t_j)}{D(z)} \leq \frac{d(v_j,t_j)+L(Q_j)}{D(v_j)-L(Q_j)} \leq \frac{d(v_j,t_j)}{D(v_j)} \left(1 + \frac{O(1)}{\log k}\right) \]

Recall that $R_j = (1 + \delta)^{g_j}$, where $g_j \sim \text{Geo}(p)$. 

Lemma (Success probability)

Assuming at least one active vertex joins $V_j$, the probability of success is at least $1 - p$.

$t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j)$

$v_j \in V_j \Rightarrow R_j \geq d(v_j, t_j)/D(v_j)$. For all $z \in S_j$,

\[
\frac{d(z,t_j)}{D(z)} \leq \frac{d(v_j,t_j)+L(Q_j)}{D(v_j)-L(Q_j)} \leq \frac{d(v_j,t_j)}{D(v_j)} \left(1 + \frac{O(1)}{\log k}\right)
\]

Recall that $R_j = (1 + \delta)g_j$, where $g_j \sim \text{Geo}(p)$. W.P. $1 - p$. 

**Lemma (Success probability)**

Assuming at least one active vertex joins $V_j$, the probability of success is at least $1 - p$.

\[ t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j) \]

$v_j \in V_j \Rightarrow R_j \geq \frac{d(v_j, t_j)}{D(v_j)}. \quad \text{For all } z \in S_j,\]

\[
\frac{d(z, t_j)}{D(z)} \leq \frac{d(v_j, t_j) + L(Q_j)}{D(v_j) - L(Q_j)} \leq \frac{d(v_j, t_j)}{D(v_j)} \left(1 + \frac{O(1)}{\log k}\right)
\]

Recall that $R_j = (1 + \delta)^{g_j}$, where $g_j \sim \text{Geo}(p)$. W.P. $1 - p$.

\[
R_j \geq (1 + \delta) \frac{d(v_j, t_j)}{D(v_j)} \geq \frac{d(z, t_j)}{D(z)}
\]
**Lemma (Success probability)**

Assuming at least one active vertex joins \( V_j \), the probability of success is at least \( 1 - p \).

\[
t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j)
\]

\[
v_j \in V_j \implies R_j \geq d(v_j, t_j)/D(v_j).
\]

For all \( z \in S_j \),

\[
\frac{d(z, t_j)}{D(z)} \leq \frac{d(v_j, t_j) + L(Q_j)}{D(v_j) - L(Q_j)} \leq \frac{d(v_j, t_j)}{D(v_j)} \left(1 + \frac{O(1)}{\log k}\right)
\]

Recall that \( R_j = (1 + \delta)^{g_j} \), where \( g_j \sim \text{Geo}(p) \). W.P. \( 1 - p \).

\[
R_j \geq (1 + \delta) \frac{d(v_j, t_j)}{D(v_j)} \geq \frac{d(z, t_j)}{D(z)}
\]
Lemma (Success probability)

Assuming at least one active vertex joins $V_j$, the probability of success is at least $1 - p$.

\[ t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j) \]

$v_j \in V_j \Rightarrow R_j \geq d(v_j, t_j)/D(v_j)$.

For all $z \in S_j$,

\[
\frac{d(z, t_j)}{D(z)} \leq \frac{d(v_j, t_j) + L(Q_j)}{D(v_j) - L(Q_j)} \leq \frac{d(v_j, t_j)}{D(v_j)} \left(1 + \frac{O(1)}{\log k}\right)
\]

Recall that $R_j = (1 + \delta)^{g_j}$, where $g_j \sim \text{Geo}(p)$. W.P. $1 - p$.

\[ R_j \geq (1 + \delta) \frac{d(v_j, t_j)}{D(v_j)} \geq \frac{d(z, t_j)}{D(z)} \]
Lemma (Success probability)

Assuming at least one active vertex joins $V_j$, the probability of success is at least $1 - p$.

$$t' \cdot D(Q) = \Theta(\log k) \cdot L(Q_j)$$

$v_j \in V_j \Rightarrow R_j \geq \frac{d(v_j, t_j)}{D(v_j)}$. For all $z \in S_j$, W.P. $1 - p$.

Recall that $R_j = (1 + \delta)g_j$, where $g_j \sim \text{Geo}(p)$.

$$R_j \geq (1 + \delta) \frac{d(v_j, t_j)}{D(v_j)} \geq \frac{d(z, t_j)}{D(z)}$$

In fact, the success probability is either 1 or $1 - p$. 
Corollary (Expected Charge)

For all $Q \in \mathcal{Q}$, $\mathbb{E}[X_Q] = O(1)$.

Proof.

Chernoff.

Arnold Filtser

Steiner Point Removal

April 26, 2018 22 / 37
Corollary (Expected Charge)

For all $Q \in Q$, $\mathbb{E}[X_Q] = O(1)$.

Proof.

$\mathbb{E}[X_Q] \leq 1 + p \cdot 2\mathbb{E}[X_Q] \implies \mathbb{E}[X_Q] \leq \frac{1}{1-2p} = O(1)$. 

$\square$
Corollary (Expected Charge)

For all $Q \in Q$, $\mathbb{E}[X_Q] = O(1)$.

Proof.

$\mathbb{E}[X_Q] \leq 1 + p \cdot 2 \mathbb{E}[X_Q] \Rightarrow \mathbb{E}[X_Q] \leq \frac{1}{1-2p} = O(1)$. 

Corollary (High Probability Charge Bound)

With high probability, for all $Q \in Q$, $X_Q = O(\log k)$. 
Corollary (Expected Charge)

For all $Q \in Q$, $\mathbb{E}[X_Q] = O(1)$. 

Proof.

$\mathbb{E}[X_Q] \leq 1 + p \cdot 2\mathbb{E}[X_Q] \Rightarrow \mathbb{E}[X_Q] \leq \frac{1}{1-2p} = O(1).$ 

Corollary (High Probability Charge Bound)

With high probability, for all $Q \in Q$, $X_Q = O(\log k)$. 

Proof.

Chernoff.
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i), \]

here \( \varphi = |Q| \).
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i), \]

here \( \varphi = |Q| \).

\( f \) is linear and monotonically increasing.
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) \text{, here } \varphi = |Q|. \]

\[ d_M(t, t') \leq d_G(t, t') + 2 \sum_{j} d_G(t_j, v_j) \]
**Definition (Charge Function)**

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) , \quad \text{here } \varphi = |Q|. \]

\[
d_M(t, t') \leq d_G(t, t') + 2 \sum_j d_G(t_j, v_j) \\
= d_G(t, t') + O(1) \cdot \sum_j D(v_j)
\]

Recall \( R_j = O(1) \), thus \( d_G(t_j, v_j) \leq R_j \cdot D(v_j) = O(D(v_j)). \)
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_{i} x_i \cdot L(Q^i) \], \quad \text{here } \varphi = |Q|.

\[
d_M(t, t') \leq d_G(t, t') + 2 \sum_{j} d_G(t_j, v_j)
\]

\[
= d_G(t, t') + O(1) \cdot \sum_{j} D(v_j)
\]

\[
= d_G(t, t') + O(\log k) \cdot \sum_{i} L(Q_j)
\]

\[
D(Q) = \Theta(\log k) \cdot L(Q)
\]

\[
L(Q) = d_G(v_a, v_b)
\]

Interval length
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) , \quad \text{here } \varphi = |Q|. \]

\[ d_M(t, t') \leq d_G(t, t') + 2 \sum_j d_G(t_j, v_j) \]

\[ = d_G(t, t') + O(1) \cdot \sum_j D(v_j) \]

\[ = d_G(t, t') + O(\log k) \cdot \sum_j L(Q_j) \]

\[ = d_G(t, t') + O(\log k) \cdot \sum_{Q \in Q} X_Q \cdot L(Q) \]
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i), \quad \text{here } \varphi = |Q|. \]

\[
d_M(t, t') \leq d_G(t, t') + 2 \sum_j d_G(t_j, v_j)
= d_G(t, t') + O(1) \cdot \sum_j D(v_j)
= d_G(t, t') + O(\log k) \cdot \sum_j L(Q_j)
= d_G(t, t') + O(\log k) \cdot \sum_{Q \in Q} X_Q \cdot L(Q)
= d_G(t, t') + O(\log k) \cdot f(X_{Q^1}, \ldots, X_{Q^\varphi})
\]
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) , \quad \text{here } \varphi = |Q|. \]

\[ d_M(t, t') = d_G(t, t') + O(\log k) \cdot f (X_{Q^1}, \ldots, X_{Q^\varphi}) \]
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\phi) = \sum_i x_i \cdot L(Q^i), \quad \text{here } \phi = |Q|. \]

\[ d_M(t, t') = d_G(t, t') + O(\log k) \cdot f(X_{Q_1}, \ldots, X_{Q_\phi}) \]

\[ \mathbb{E}[f(X_{Q_1}, \ldots, X_{Q_\phi})] = \sum_{Q \in Q} \mathbb{E}[X_Q] \cdot L(Q) \]
**Definition (Charge Function)**

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) , \quad \text{here } \varphi = |Q|. \]

\[ d_M(t, t') = d_G(t, t') + O(\log k) \cdot f(X_{Q^1}, \ldots, X_{Q^\varphi}) \]

\[ \mathbb{E}[f(X_{Q^1}, \ldots, X_{Q^\varphi})] = \sum_{Q \in \mathcal{Q}} \mathbb{E}[X_Q] \cdot L(Q) = O(1) \cdot \sum_{Q \in \mathcal{Q}} L(Q) = O(1) \cdot d_G(t, t') \]
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) \, , \quad \text{here } \varphi = |Q|. \]

\[ d_M(t, t') = d_G(t, t') + O(\log k) \cdot f(X_{Q^1}, \ldots, X_{Q^\varphi}) \]

\[ \mathbb{E}[f(X_{Q^1}, \ldots, X_{Q^\varphi})] = \sum_{Q \in Q} \mathbb{E}[X_Q] \cdot L(Q) \]
\[ = O(1) \cdot \sum_{Q \in Q} L(Q) = O(1) \cdot d_G(t, t') \]

Theorem

The expected distortion of the minor \( M \) returned by the Noisy Voronoi algorithm is \( O(\log k) \).
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) , \quad \text{here } \varphi = |Q|. \]

\[ d_M(t, t') = d_G(t, t') + O(\log k) \cdot f(X_{Q^1}, \ldots, X_{Q^\varphi}) \]

Moreover, with high probability
**Definition (Charge Function)**

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) , \quad \text{here } \varphi = |Q| . \]

\[ d_M(t, t') = d_G(t, t') + O(\log k) \cdot f(X_{Q^1}, \ldots, X_{Q^\varphi}) \]

Moreover, with high probability

\[ f(X_{Q^1}, \ldots, X_{Q^\varphi}) = \sum_{Q \in \mathcal{Q}} X_Q \cdot L(Q) \]
Definition (Charge Function)

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) , \quad \text{here } \varphi = |Q|. \]

\[ d_M(t, t') = d_G(t, t') + O(\log k) \cdot f(X_{Q^1}, \ldots, X_{Q^\varphi}) \]

Moreover, with high probability

\[ f(X_{Q^1}, \ldots, X_{Q^\varphi}) = \sum_{Q \in Q} X_Q \cdot L(Q) \]

\[ = O(\log k) \cdot \sum_{Q \in Q} L(Q) = O(\log k) \cdot d_G(t, t') \]
**Definition (Charge Function)**

\[ f(x_1, x_2, \ldots, x_\varphi) = \sum_i x_i \cdot L(Q^i) \text{, here } \varphi = |Q|. \]

\[ d_M(t, t') = d_G(t, t') + O(\log k) \cdot f(X_{Q^1}, \ldots, X_{Q^\varphi}) \]

Moreover, with high probability

\[ f(X_{Q^1}, \ldots, X_{Q^\varphi}) = \sum_{Q \in Q} X_Q \cdot L(Q) \]

\[ = O(\log k) \cdot \sum_{Q \in Q} L(Q) = O(\log k) \cdot d_G(t, t') \]

**Theorem**

*With high probability*, the *Noisy Voronoi algorithm* returns a minor \( M \) with distortion \( O(\log^2 k) \).
But you promised distortion $O(\log k)$!
Analyze $f(X_{Q^1}, \ldots, X_{Q^\varphi}) = \sum_{Q \in Q} X_Q \cdot L(Q)$ better.
Analyze $f(X_{Q_1}, \ldots, X_{Q_\varphi}) = \sum_{Q \in Q} X_Q \cdot L(Q)$ better.  
But $X_{Q_1}, \ldots, X_{Q_\varphi}$ are dependent.
Analyze $f(X_{Q_1}, \ldots, X_{Q_{\varphi}}) = \sum_{Q \in Q} X_Q \cdot L(Q)$ better.

But $X_{Q_1}, \ldots, X_{Q_{\varphi}}$ are dependent. What can we do?
Analyze $f(X_{Q^1}, \ldots, X_{Q^\varphi}) = \sum_{Q \in \mathcal{Q}} X_Q \cdot L(Q)$ better.

But $X_{Q^1}, \ldots, X_{Q^\varphi}$ are dependent. What can we do?

They maybe dependent, but in a “positive” way!
Idea

We will introduce new **series** of independent **random variables** and show that they **dominate** \( X_{Q^1}, \ldots, X_{Q^\varphi} \).
We will introduce new **series** of independent random variables and show that they **dominate** $X_{Q^1}, \ldots, X_{Q^\varphi}$. 

![Active vs Inactive](image-url)
Idea

We will introduce new series of independent random variables and show that they dominate $X_{Q^1}, \ldots, X_{Q^\phi}$. 

Active

Inactive
Idea

We will introduce new **series** of independent random variables and show that they **dominate** $X_{Q^1}, \ldots, X_{Q^φ}$. 

![Active and Inactive Buckets](image-url)
Denote by $A(B)$ the number of active Coins in the bucket $B$. Denote by $IN(B)$ the number of inactive Coins in the bucket $B$. 
Coupling

$\#S(Q^1) = 1$
$X_{Q^1} = 0$

$\#S(Q^{i-1}) = 1$
$X_{Q^{i-1}} = 0$

$\#S(Q^i) = 1$
$X_{Q^i} = 0$

$\#S(Q^{i+1}) = 1$
$X_{Q^{i+1}} = 0$

$\#S(Q^\phi) = 1$
$X_{Q^\phi} = 0$

$Q^1$

$Q^{i-1}$

$Q^i$

$Q^{i+1}$

$Q^\phi$

$B_1$

$B_{i-1}$

$B_i$

$B_{i+1}$

$B_{\phi}$
Coupling

\[ \#S(Q^1) = 1 \quad X_{Q^1} = 0 \]

\[ \#S(Q^{i-1}) = 1 \quad X_{Q^{i-1}} = 0 \]

\[ \#S(Q^i) = 1 \quad X_{Q^i} = 0 \]

\[ \#S(Q^{i+1}) = 1 \quad X_{Q^{i+1}} = 0 \]

\[ \#S(Q^\varphi) = 1 \quad X_{Q^\varphi} = 0 \]

\[ B_1, \ldots, B_\varphi \text{ are independent buckets.} \]
$\#S(Q^1) = 1 \quad X_{Q^1} = 0$

$\#S(Q^{i-1}) = 1 \quad X_{Q^{i-1}} = 0$

$\#S(Q^i) = 1 \quad X_{Q^i} = 0$

$\#S(Q^{i+1}) = 1 \quad X_{Q^{i+1}} = 0$

$\#S(Q^\phi) = 1 \quad X_{Q^\phi} = 0$

$B_1, \ldots , B_\phi$ are independent buckets.

We execute Noisy Voronoi algorithm and use it in order to determine $IN(B_1), \ldots , IN(B_\phi)$. 
Maintain, for all $i$, 

$$X_{Q^i} \leq IN(B_i) \quad \& \quad \#S(Q^i) \leq A(B_i)$$
Coupling

**Maintain**, for all \(i\), \(X_{Q_i} \leq IN(B_i) \land \#S(Q_i) \leq A(B_i)\)
Coupling

**Maintain**, for all $i$, $X_{Q_i} \leq \text{IN}(B_i)$ & $\#S(Q^i) \leq A(B_i)$

Suppose $t_j$ grows cluster $V_j$. 
Coupling

**Maintain**, for all $i$, $X_{Q^i} \leq IN(B_i)$ & $\#S(Q^i) \leq A(B_i)$

Suppose $t_j$ grows cluster $V_j$.

- If no active vertex joins $V_j$. **Nothing change.**
Coupling

**Maintain**, for all $i$, $X_{Q_i} \leq \text{IN}(B_i)$ & $\#S(Q^i) \leq A(B_i)$

Suppose $t_j$ grows cluster $V_j$.

- If **no** active vertex **joins** $V_j$. **Nothing change.**

Else, $v_j \in S_j \subseteq Q_j$ is the first vertex to **join** $V_j$. $B_{(j)}$ is the corresponding bucket to $Q_j$. 

$\text{IN}(B_{(j)}) \leftarrow \text{IN}(B_{(j)}) + 1$, $\text{A}(B_{(j)}) \leftarrow \text{A}(B_{(j)}) + 1$.
Coupling

**Maintain**, for all $i$, $X_{Q_i} \leq IN(B_i)$ & $\#S(Q^i) \leq A(B_i)$

Suppose $t_j$ grows cluster $V_j$.

- If no active vertex joins $V_j$. **Nothing change.**

Else, $v_j \in S_j \subseteq Q_j$ is the first vertex to join $V_j$.
$B_{(j)}$ is the corresponding bucket to $Q_j$.
Let $p'$ be the **probability** that **not all** of $S_j$ joins $V_j$. Recall $p' \leq p$. 
Coupling

**Maintain**, for all \( i \), \( X_{Q_i} \leq IN(B_i) \) & \( \#S(Q_i) \leq A(B_i) \)

Suppose \( t_j \) grows cluster \( V_j \).

- If **no** active vertex **joins** \( V_j \). **Nothing change.**

Else, \( v_j \in S_j \subseteq Q_j \) is the first vertex to **join** \( V_j \).

\( B(j) \) is the corresponding bucket to \( Q_j \).

Let \( p' \) be the **probability** that **not all** of \( S_j \) **joins** \( V_j \). Recall \( p' \leq p \).

- If not all of \( S_j \) **joins** \( V_j \): **Fail in both processes.**
  
  Add two active coins.
Coupling

**Maintain**, for all $i$, $X_{Q_i} \leq IN(B_i) \& \#S(Q^i) \leq A(B_i)$

Suppose $t_j$ grows cluster $V_j$.

- If **no** active vertex joins $V_j$. **Nothing change.**

Else, $v_j \in S_j \subseteq Q_j$ is the first vertex to **join** $V_j$.

$B(j)$ is the corresponding bucket to $Q_j$.

Let $p'$ be the **probability** that not all of $S_j$ joins $V_j$. Recall $p' \leq p$.

- If not all of $S_j$ joins $V_j$: **Fail in both processes**.
  Add two active coins.
  
  - $A(B(j)) \leftarrow A(B(j)) + 1$, $IN(B(j)) \leftarrow IN(B(j)) + 1$.
  
  For $i \neq (j)$, $A(B_i), IN(B_i)$ unchanged.
Coupling

**Maintain**, for all $i$, $X_{Q_i} \leq \text{IN}(B_i)$ & $\#S(Q^i) \leq A(B_i)$

Suppose $t_j$ grows cluster $V_j$.

- If no active vertex joins $V_j$. *Nothing change.*

Else, $v_j \in S_j \subseteq Q_j$ is the first vertex to join $V_j$.

$B_{(j)}$ is the corresponding bucket to $Q_j$.

Let $p'$ be the **probability** that not all of $S_j$ joins $V_j$. Recall $p' \leq p$.

- If not all of $S_j$ joins $V_j$: *Fail in both processes.*
  
  Add two active coins.

  - $A(B_{(j)}) \leftarrow A(B_{(j)}) + 1$, $\text{IN}(B_{(j)}) \leftarrow \text{IN}(B_{(j)}) + 1$.
    
    For $i \neq (j)$, $A(B_i)$, $\text{IN}(B_i)$ unchanged.

  - $\#S(Q_j) \leq \#S(Q_j) + 1$, $X_{Q_j} \leq X_{Q_j} + 1$.
    
    For $i \neq j$, $\#S(Q_i)$, $X_{Q_i}$ might only decrease.
**Coupling**

**Maintain**, for all \(i\), \(X_{Q_i} \leq \text{IN}(B_i) \land \#S(Q_i) \leq A(B_i)\)

Suppose \(t_j\) grows cluster \(V_j\).

- If no active vertex joins \(V_j\). Nothing change.

Else, \(v_j \in S_j \subseteq Q_j\) is the first vertex to join \(V_j\).

\(B_{(j)}\) is the corresponding bucket to \(Q_j\).

Let \(p'\) be the **probability** that not all of \(S_j\) joins \(V_j\). Recall \(p' \leq p\).
Coupling

**Maintain**, for all $i$, $X_{Q_i} \leq IN(B_i)$ & $\#S(Q_i) \leq A(B_i)$

Suppose $t_j$ grows cluster $V_j$.

- If **no** active vertex joins $V_j$. **Nothing change.**

Else, $v_j \in S_j \subseteq Q_j$ is the first vertex to **join** $V_j$.

$B_{(j)}$ is the corresponding bucket to $Q_j$.

Let $p'$ be the **probability** that **not all** of $S_j$ joins $V_j$. Recall $p' \leq p$.

- If all of $S_j$ joins $V_j$: **Success in alg.**
  
  With probability $\frac{p - p'}{1 - p'}$, add two active coins (**fail in buckets**).
**Coupling**

**Maintain**, for all $i$, $X_{Q_i} \leq \text{IN}(B_i)$ & $\#S(Q_i) \leq A(B_i)$

Suppose $t_j$ grows cluster $V_j$.

- If no active vertex joins $V_j$. **Nothing change.**

Else, $v_j \in S_j \subseteq Q_j$ is the first vertex to join $V_j$.

$B(j)$ is the corresponding bucket to $Q_j$.

Let $p'$ be the **probability** that **not all** of $S_j$ joins $V_j$. Recall $p' \leq p$.

- If all of $S_j$ joins $V_j$: **Success in alg.**

  With probability $\frac{p - p'}{1 - p'}$, add two active coins (fail in buckets).

    - $A(B(j)) \geq A(B(j)) - 1$, $\text{IN}(B(j)) \leftarrow \text{IN}(B(j)) + 1$.

      For $i \neq (j)$, $A(B_i)$, $\text{IN}(B_i)$ unchanged.
**Coupling**

**Maintain**, for all \(i\), \(X_{Q_i} \leq IN(B_i) \& \#S(Q_i) \leq A(B_i)\)

Suppose \(t_j\) grows cluster \(V_j\).

- If **no** active vertex joins \(V_j\). **Nothing change.**

Else, \(v_j \in S_j \subseteq Q_j\) is the first vertex to join \(V_j\).

\(B(j)\) is the corresponding bucket to \(Q_j\).

Let \(p'\) be the **probability** that **not all** of \(S_j\) joins \(V_j\). Recall \(p' \leq p\).

- If all of \(S_j\) joins \(V_j\): **Success in alg.**
  
  With probability \(\frac{p-p'}{1-p'}\), add two active coins (fail in buckets).
  
  - \(A(B(j)) \geq A(B(j)) - 1\), \(IN(B(j)) \leftarrow IN(B(j)) + 1\).
    For \(i \neq (j)\), \(A(B_i), IN(B_i)\) unchanged.
  
  - \(#S(Q_j) \leq #S(Q_j) - 1\), \(X_{Q_j} \leq X_{Q_j} + 1\).
    For \(i \neq j\), \(#S(Q_i), X_{Q_i}\) might only decrease.
**Coupling**

**Maintain**, for all $i$, $X_{Q_i} \leq \text{IN}(B_i)$ & $\#S(Q_i) \leq \text{A}(B_i)$

Suppose $t_j$ grows cluster $V_j$.

- If no active vertex joins $V_j$. Nothing change.

Else, $v_j \in S_j \subseteq Q_j$ is the first vertex to join $V_j$. $B_{(j)}$ is the corresponding bucket to $Q_j$.

Let $p'$ be the probability that not all of $S_j$ joins $V_j$. Recall $p' \leq p$.

- If all of $S_j$ joins $V_j$: Success in alg.
  
  With probability $\frac{p - p'}{1 - p'}$, add two active coins (fail in buckets).

  - $A(B_{(j)}) \geq A(B_{(j)}) - 1$, $\text{IN}(B_{(j)}) \leftarrow \text{IN}(B_{(j)}) + 1$.
    For $i \neq (j)$, $A(B_i)$, $\text{IN}(B_i)$ unchanged.
  
  - $\#S(Q_j) \leq \#S(Q_j) - 1$, $X_{Q_j} \leq X_{Q_j} + 1$.
    For $i \neq j$, $\#S(Q_i)$, $X_{Q_i}$ might only decrease.

The probability of failure in the bucket is: $p' + (1 - p') \cdot \frac{p - p'}{1 - p'} = p$
**Coupling**

**Maintain**, for all $i$, $X_{Q_i} \leq IN(B_i) \& \#S(Q_i) \leq A(B_i)$

Suppose $t_j$ grows cluster $V_j$.

- If no active vertex joins $V_j$. **Nothing change.**

Else, $v_j \in S_j \subseteq Q_j$ is the first vertex to join $V_j$.

$B(j)$ is the corresponding bucket to $Q_j$.

Let $p'$ be the **probability** that **not all** of $S_j$ joins $V_j$. Recall $p' \leq p$.

- If all of $S_j$ joins $V_j$: **Success in alg.**

  With probability $\frac{p-p'}{1-p'}$, add two active coins (**fail in buckets**).

  - $A(B(j)) \geq A(B(j)) - 1$, $IN(B(j)) \leftarrow IN(B(j)) + 1$.
    For $i \neq (j)$, $A(B_i)$, $IN(B_i)$ unchanged.
  
  - $\#S(Q_j) \leq \#S(Q_j) - 1$, $X_{Q_j} \leq X_{Q_j} + 1$.
    For $i \neq j$, $\#S(Q_i)$, $X_{Q_i}$ might only decrease.

The probability of failure in the bucket is: $p' + (1 - p') \cdot \frac{p-p'}{1-p'} = p$

The **marginal distribution** on the buckets is correct!
While the processes remain coupled, we maintained for all $i$, 

$$X_{Q^i} \leq IN(B_i) \quad \& \quad \#S(Q^i) \leq A(B_i)$$
While the processes remain coupled, we maintained for all $i$,

$$X_{Qi} \leq \text{IN}(B_i) \quad \& \quad \#S(Q^i) \leq A(B_i)$$

At end, if active coins remain, just flip them regularly.
While the processes remain coupled, we maintained for all $i$,

$$X_{Q^i} \leq IN(B_i) \quad \& \quad \#S(Q^i) \leq A(B_i)$$

At end, if active coins remain, just flip them regularly.

$IN(B)$ can only grow!
While the processes remain coupled, we maintained for all $i$,

$$X_{Q^i} \leq IN(B_i) \quad \& \quad \#S(Q^i) \leq A(B_i)$$

At end, if active coins remain, just flip them regularly. \textit{IN(}$B\textit{)} can only grow!

Thus, \textit{($X_{Q^1}, \ldots, X_{Q^\varphi}$)} $\leq$ \textit{(}$IN(B_1), \ldots, IN(B_\varphi)$\textit{)} \textbf{coordinatewise}
While the processes remain coupled, we maintained for all $i$,

$$X_{Q^i} \leq IN(B_i) \quad \& \quad \#S(Q^i) \leq A(B_i)$$

At end, if active coins remain, just flip them regularly. $IN(B)$ can only grow!

Thus, $(X_{Q^1}, \ldots, X_{Q^\varphi}) \leq (IN(B_1), \ldots, IN(B_\varphi))$ coordinatewise

**Corollary (The buckets dominate the detour charges)**

For all $\alpha \geq 0$,

$$\Pr[f(X_{Q^1}, \ldots, X_{Q^\varphi}) \geq \alpha] \leq \Pr[f(IN(B_1), \ldots, IN(B_\varphi)) \geq \alpha]$$
Lemma (Exponential Distribution Dominates Bucket)

For all $\alpha \geq 0$,

$$\Pr[\text{IN} \geq \alpha] \leq \Pr[\text{Exp}(10) + 1 \geq \alpha]$$
Lemma (Exponential Distribution Dominates Bucket)

For all $\alpha \geq 0$,

$$\Pr[\text{IN}(B) \geq \alpha] \leq \Pr[\text{Exp}(10) + 1 \geq \alpha]$$

Proof.

Meh. Too Technical.
Lemma (Exponential Distribution Dominates Bucket)

For all $\alpha \geq 0$,

$$\Pr [\text{IN}(B) \geq \alpha] \leq \Pr [\text{Exp}(10) + 1 \geq \alpha]$$

Corollary (Series of Exponential Dominates the Buckets)

For all $\alpha \geq 0$,

$$\Pr [f (\text{IN}(B_1), \ldots, \text{IN}(B_\varphi)) \geq \alpha]$$

$$\leq \Pr [f (\text{Exp}(10) + 1, \ldots, \text{Exp}(10) + 1) \geq \alpha]$$
Lemma (**Exponential Distribution Dominates Bucket**)

For all $\alpha \geq 0$,

$$\Pr[\text{IN}(B) \geq \alpha] \leq \Pr[\text{Exp}(10) + 1 \geq \alpha]$$

Corollary (**Series of Exponential Dominates the Buckets**)

For all $\alpha \geq 0$,

$$\Pr[f(\text{IN}(B_1), \ldots, \text{IN}(B_\varphi)) \geq \alpha]$$

$$\leq \Pr[f(\text{Exp}(10) + 1, \ldots, \text{Exp}(10) + 1) \geq \alpha]$$

Proof.

You know the drill... ($f$ is linear and monotone coordinatewise.)
Lemma (Exponential Distribution Dominates Bucket)

For all $\alpha \geq 0$,

$$\Pr \left[ \text{IN}(B) \geq \alpha \right] \leq \Pr \left[ \text{Exp}(10) + 1 \geq \alpha \right]$$

Corollary (Series of Exponential Dominates the Buckets)

For all $\alpha \geq 0$,

$$\Pr \left[ f(\text{IN}(B_1), \ldots, \text{IN}(B_\varphi)) \geq \alpha \right] \leq \Pr \left[ f(\text{Exp}(10) + 1, \ldots, \text{Exp}(10) + 1) \geq \alpha \right]$$

Note that

$$f(\text{Exp}(10) + 1, \ldots, \text{Exp}(10) + 1) = f(\text{Exp}(10), \ldots, \text{Exp}(10)) + f(1, \ldots, 1)$$
Lemma (Exponential Distribution Dominates Bucket)

For all $\alpha \geq 0$,

$$\Pr[\text{IN}(B) \geq \alpha] \leq \Pr[\text{Exp}(10) + 1 \geq \alpha]$$

Corollary (Series of Exponential Dominates the Buckets)

For all $\alpha \geq 0$,

$$\Pr[f(\text{IN}(B_1), \ldots, \text{IN}(B_{\varphi})) \geq \alpha]$$

$$\leq \Pr[f(\text{Exp}(10) + 1, \ldots, \text{Exp}(10) + 1) \geq \alpha]$$

Thus, in order to bound $f(X_{Q_1}, \ldots, X_{Q_{\varphi}})$ it will be enough to bound

$$f(\text{Exp}(10), \ldots, \text{Exp}(10)) = \sum_{i=1}^{\varphi} \text{Exp}(10) \cdot L(Q_i)$$

$$= \sum_{i=1}^{\varphi} \text{Exp}(10 \cdot L(Q_i))$$
Goal: bound $\sum_{i=1}^{\varphi} \text{Exp} (10 \cdot L(Q_i))$. 
Goal: bound $\sum_{i=1}^{\phi} \text{Exp} (10 \cdot L(Q_i))$.

Lemma (Concentration Bound for Exp)

Let $X_1, \ldots, X_n$ be independent random variables, where $X_i \sim \text{Exp}(\lambda_i)$.

Set: $X = \sum_i X_i$, $\lambda_M = \max_i \lambda_i$, $\mu = \mathbb{E}[X] = \sum_i \lambda_i$. 

For $a \geq 2 \mu$, $\Pr[X \geq a] \leq \exp(-\frac{1}{2} \lambda_M (a - 2\mu))$. 

In our case, $X_i \sim \text{Exp}(10 \cdot L(Q_i))$, $X = \sum_i X_i$. 

$\mu = \mathbb{E}[X] = \sum_i \lambda_i$. 

$\lambda_M = \max_i \{10 \cdot L(Q_i)\} = O(d_G(t, t')) \log k)$. 

Arnold Filtser  
Steiner Point Removal  
April 26, 2018  
34 / 37
Goal: bound $\sum_{i=1}^{\varphi} \text{Exp} \left( 10 \cdot L(Q_i) \right)$.

Lemma (Concentration Bound for Exp)

$X_1, \ldots, X_n$ are i.r.v, where $X_i \sim \text{Exp}(\lambda_i)$.

Set: $X = \sum_i X_i$, $\lambda_M = \max_i \lambda_i$, $\mu = \mathbb{E}[X] = \sum_i \lambda_i$.

For $a \geq 2\mu$ $\Pr[X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right)$
Goal: bound $\sum_{i=1}^{\varphi} \operatorname{Exp}(10 \cdot L(Q_i))$.

**Lemma (Concentration Bound for Exp)**

$X_1, \ldots, X_n$ are i.r.v, where $X_i \sim \operatorname{Exp}(\lambda_i)$.

Set: $X = \sum_i X_i, \quad \lambda_M = \max_i \lambda_i, \quad \mu = \mathbb{E}[X] = \sum_i \lambda_i$.

For $a \geq 2\mu$ \quad $\Pr[X \geq a] \leq \exp\left(-\frac{1}{2\lambda_M} (a - 2\mu)\right)$

In our case, $X_i \sim \operatorname{Exp}(10 \cdot L(Q_i))$. $X = \sum_i X_i$. 
Goal: bound \( \sum_{i=1}^{\varphi} \exp(10 \cdot L(Q_i)) \).

Lemma (Concentration Bound for Exp)

\( X_1, \ldots, X_n \) are i.r.v, where \( X_i \sim \exp(\lambda_i) \).

Set: \( X = \sum_i X_i, \quad \lambda_M = \max_i \lambda_i, \quad \mu = \mathbb{E}[X] = \sum_i \lambda_i \).

For \( a \geq 2\mu \) \( \Pr[X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right) \)

In our case, \( X_i \sim \exp(10 \cdot L(Q_i)) \). \( X = \sum_i X_i \).

\[
\mu = \mathbb{E}[X] = \mathbb{E} \left[ \sum_i X_i \right] = \sum_i \mathbb{E}[X_i] = \sum_i 10 \cdot L(Q_i) \leq 10 \cdot d_G(t, t')
\]
Goal: bound $\sum_{i=1}^{\varphi} \text{Exp} \left( 10 \cdot L(Q_i) \right)$.

Lemma (Concentration Bound for Exp)

Let $X_1, \ldots, X_n$ be i.r.v, where $X_i \sim \text{Exp}(\lambda_i)$.

Set: $X = \sum_i X_i$, $\lambda_M = \max_i \lambda_i$, $\mu = \mathbb{E}[X] = \sum_i \lambda_i$.

For $a \geq 2\mu$ \hspace{1cm} $\Pr[X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right)$

In our case, $X_i \sim \text{Exp}(10 \cdot L(Q_i))$. $X = \sum_i X_i$.

\[
\mu = \mathbb{E}[X] = \mathbb{E} \left[ \sum_i X_i \right] = \sum_i \mathbb{E}[X_i] = \sum_i 10 \cdot L(Q_i) \leq 10 \cdot d_G(t, t')
\]

$\lambda_M = \max_i \{10 \cdot L(Q_i)\} = \max_i \left\{ O \left( \frac{D(Q_i)}{\log k} \right) \right\} = O \left( \frac{d_G(t, t')}{\log k} \right)$
\[ \mu \leq 10 \cdot d_G(t, t') \]

Thus for \( a = 30 \cdot d_G(t, t') \)

\[
\Pr [X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right) =
\]

\[ \lambda_M = O \left( \frac{d_G(t, t')}{\log k} \right) \]
\[ \mu \leq 10 \cdot d_G(t, t') \]

Thus for \( a = 30 \cdot d_G(t, t') \)

\[
Pr [X \geq a] \leq \exp \left( - \frac{1}{2\lambda_M} (a - 2\mu) \right) = \exp (\Omega (\log k)) = \frac{1}{k^3}
\]

\[
\lambda_M = O \left( \frac{d_G(t, t')}{\log k} \right)
\]
Thus for $a = 30 \cdot d_G(t, t')$

$$\Pr [X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M}(a - 2\mu) \right) = \exp (\Omega (\log k)) = \frac{1}{k^3}$$

We conclude

$$\Pr \left[ f (X_{Q_1}, \ldots, X_{Q_\varphi}) \geq O(d_G(t, t')) \right]$$
\[ \mu \leq 10 \cdot d_G(t, t') \]

Thus for \( a = 30 \cdot d_G(t, t') \)

\[
\Pr [X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right) = \exp (\Omega (\log k)) = \frac{1}{k^3}
\]

We conclude

\[
\Pr \left[ f (X_{Q_1}, \ldots, X_{Q_\varphi}) \geq O(d_G(t, t')) \right] \\
\leq \Pr \left[ f (\text{IN}(B_1), \ldots, \text{IN}(B_\varphi)) \geq O(d_G(t, t')) \right]
\]
\[ \mu \leq 10 \cdot d_G(t, t') \]

Thus for \( a = 30 \cdot d_G(t, t') \)

\[ \Pr [X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right) = \exp (\Omega (\log k)) = \frac{1}{k^3} \]

We conclude

\[
\begin{align*}
\Pr & \left[ f \left( X_{Q_1}, \ldots, X_{Q_{\varphi}} \right) \geq O(d_G(t, t')) \right] \\
& \leq \Pr \left[ f \left( IN(B_1), \ldots, IN(B_{\varphi}) \right) \geq O(d_G(t, t')) \right] \\
& \leq \Pr \left[ f \left( \text{Exp}(10), \ldots, \text{Exp}(10) \right) \geq O(d_G(t, t')) \right]
\end{align*}
\]
\[ \mu \leq 10 \cdot d_G(t, t') \]

Thus for \( a = 30 \cdot d_G(t, t') \)

\[
\Pr [X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right) = \exp (\Omega (\log k)) = \frac{1}{k^3}
\]

We conclude

\[
\Pr \left[ f \left( X_{Q_1}, \ldots, X_{Q_\varphi} \right) \geq O(d_G(t, t')) \right] \\
\leq \Pr \left[ f \left( \text{IN}(B_1), \ldots, \text{IN}(B_\varphi) \right) \geq O(d_G(t, t')) \right] \\
\leq \Pr \left[ f \left( \text{Exp}(10), \ldots, \text{Exp}(10) \right) \geq O(d_G(t, t')) \right] \\
= \Pr [X \geq a] \leq \frac{1}{k^3}
\]
\( \mu \leq 10 \cdot d_G(t, t') \)

Thus for \( a = 30 \cdot d_G(t, t') \)

\[
\Pr [X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right) = \exp (\Omega (\log k)) = \frac{1}{k^3}
\]

We conclude

\[
\Pr \left[ f \left( X_{Q_1}, \ldots, X_{Q_\varphi} \right) \geq O(d_G(t, t')) \right] \\
\leq \Pr \left[ f \left( \text{IN}(B_1), \ldots, \text{IN}(B_\varphi) \right) \geq O(d_G(t, t')) \right] \\
\leq \Pr \left[ f \left( \text{Exp}(10), \ldots, \text{Exp}(10) \right) \geq O(d_G(t, t')) \right] \\
= \Pr [X \geq a] \leq \frac{1}{k^3}
\]

If this event indeed occurs
\[ \mu \leq 10 \cdot d_G(t, t') \]

Thus for \( a = 30 \cdot d_G(t, t') \)

\[ \Pr[X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right) = \exp (\Omega (\log k)) = \frac{1}{k^3} \]

We conclude

\[ \Pr[f (X_{Q_1}, \ldots, X_{Q_{\varphi}}) \geq O(d_G(t, t'))] \]
\[ \leq \Pr[f (IN(B_1), \ldots, IN(B_{\varphi})) \geq O(d_G(t, t'))] \]
\[ \leq \Pr[f (\text{Exp}(10), \ldots, \text{Exp}(10)) \geq O(d_G(t, t'))] \]
\[ = \Pr[X \geq a] \leq \frac{1}{k^3} \]

If this event indeed occurs

\[ d_M(t, t') \leq d_G(t, t') + O(\log k) \cdot f (X_{Q_1}, \ldots, X_{Q_{\varphi}}) \]
\[ \mu \leq 10 \cdot d_G(t, t') \]

Thus for \( a = 30 \cdot d_G(t, t') \)

\[
\Pr[X \geq a] \leq \exp \left( -\frac{1}{2\lambda_M} (a - 2\mu) \right) = \exp (\Omega (\log k)) = \frac{1}{k^3}
\]

We conclude

\[
\Pr \left[ f \left( X_{Q_1}, \ldots, X_{Q_\varphi} \right) \geq O(d_G(t, t')) \right] \\
\leq \Pr \left[ f \left( \text{IN}(B_1), \ldots, \text{IN}(B_\varphi) \right) \geq O(d_G(t, t')) \right] \\
\leq \Pr \left[ f \left( \text{Exp}(10), \ldots, \text{Exp}(10) \right) \geq O(d_G(t, t')) \right] \\
= \Pr[X \geq a] \leq \frac{1}{k^3}
\]

If this event indeed occurs

\[
d_M(t, t') \leq d_G(t, t') + O(\log k) \cdot f \left( X_{Q_1}, \ldots, X_{Q_\varphi} \right) \\
= O(\log k) \cdot d_G(t, t')
\]
By union bound, w.h.p for all $t, t'$, $d_M(t, t') = O(\log k) \cdot d_G(t, t')$. 
By union bound, w.h.p for all \( t, t' \), \( d_M(t, t') = O(\log k) \cdot d_G(t, t') \).
Open Question

Close the gap between 8 to log $k$!
Open Question

Close the gap between $8$ to $\log k$!

Thank You!
We can assume that edges has infinitesimally small weights. Otherwise we simply subdivide.

The set of minors and the geometry of the terminals remain the same!
Algorithm 1 $M = \text{Ball-Growing}(G = (V, E), w, K = \{t_1, \ldots, t_k\})$

1. Set $r \leftarrow 1 + \delta / \ln k$, where $\delta = 1/80$.
2. Set $D \leftarrow \frac{\delta}{\ln k}$.
3. For each $j \in [k]$, set $V_j \leftarrow \{t_j\}$, and set $R_j \leftarrow 0$.
4. Set $V_\perp \leftarrow V \setminus \left( \bigcup_{j=1}^{k} V_j \right)$.
5. Set $\ell \leftarrow 0$.
6. While $\left( \bigcup_{j=1}^{k} V_j \right) \neq V$ do
   7. For $j$ from 1 to $k$ do
      8. Choose independently at random $q_j^\ell$ distributed according to $\text{Exp}(D \cdot r^\ell)$.
      9. Set $R_j \leftarrow R_j + q_j^\ell$.
     10. Set $V_j \leftarrow B_{G[V_\perp \cup V_j]}(t_j, R_j)$.
     11. Set $V_\perp \leftarrow V \setminus \left( \bigcup_{j=1}^{k} V_j \right)$.
   8. End for
   9. $\ell \leftarrow \ell + 1$.
10. End while
11. Return the terminal-centered minor $M$ of $G$ induced by $V_1, \ldots, V_k$. 
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 0 \]
\[ R_2 = 0 \]
\[ R_3 = 0 \]
\[ R_4 = 0 \]
\[ R_5 = 0 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 0.2 \]
\[ R_2 = 0 \]
\[ R_3 = 0 \]
\[ R_4 = 0 \]
\[ R_5 = 0 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 0.2 \]
\[ R_2 = 0.1 \]
\[ R_3 = 0 \]
\[ R_4 = 0 \]
\[ R_5 = 0 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 0.2 \]
\[ R_2 = 0.1 \]
\[ R_3 = 0.3 \]
\[ R_4 = 0.1 \]
\[ R_5 = 0.25 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 0.5 \]
\[ R_2 = 0.1 \]
\[ R_3 = 0.3 \]
\[ R_4 = 0.1 \]
\[ R_5 = 0.25 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 0.5 \]
\[ R_2 = 0.55 \]
\[ R_3 = 0.3 \]
\[ R_4 = 0.1 \]
\[ R_5 = 0.25 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 0.5 \]
\[ R_2 = 0.55 \]
\[ R_3 = 0.6 \]
\[ R_4 = 0.2 \]
\[ R_5 = 0.8 \]
Ball
Growing
Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 0.9 \]
\[ R_2 = 0.55 \]
\[ R_3 = 0.6 \]
\[ R_4 = 0.2 \]
\[ R_5 = 0.8 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\( R_1 = 0.9 \)
\( R_2 = 1.05 \)
\( R_3 = 0.6 \)
\( R_4 = 0.2 \)
\( R_5 = 0.8 \)
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 0.9 \]
\[ R_2 = 1.05 \]
\[ R_3 = 0.85 \]
\[ R_4 = 0.7 \]
\[ R_5 = 1.1 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[
R_1 = 1.1 \\
R_2 = 1.05 \\
R_3 = 0.85 \\
R_4 = 0.7 \\
R_5 = 1.1
\]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[
R_1 = 1.1 \\
R_2 = 1.2 \\
R_3 = 0.85 \\
R_4 = 0.7 \\
R_5 = 1.1
\]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 1.1 \]
\[ R_2 = 1.2 \]
\[ R_3 = 1.1 \]
\[ R_4 = 1.05 \]
\[ R_5 = 1.9 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 2.5 \]
\[ R_2 = 2.2 \]
\[ R_3 = 2.3 \]
\[ R_4 = 1.8 \]
\[ R_5 = 2.8 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 2.9 \]
\[ R_2 = 3.2 \]
\[ R_3 = 3.15 \]
\[ R_4 = 2.2 \]
\[ R_5 = 3.2 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

$R_1 = 3.4$

$R_2 = 4.1$

$R_3 = 3.8$

$R_4 = 3.1$

$R_5 = 3.6$
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 4.2 \]
\[ R_2 = 4.8 \]
\[ R_3 = 4.5 \]
\[ R_4 = 3.7 \]
\[ R_5 = 3.8 \]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

$R_1 = 5.5$
$R_2 = 6$
$R_3 = 4.9$
$R_4 = 4.5$
$R_5 = 5.1$
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[
\begin{align*}
R_1 &= 5.5 \\
R_2 &= 6 \\
R_3 &= 4.9 \\
R_4 &= 4.5 \\
R_5 &= 5.1
\end{align*}
\]
Ball Growing Algorithm

Arbitrary order.

Expand cluster in every round.

\[ R_1 = 5.5 \]
\[ R_2 = 6 \]
\[ R_3 = 4.9 \]
\[ R_4 = 4.5 \]
\[ R_5 = 5.1 \]
Algorithm 2: $M = \text{Noisy-Voronoi}(G = (V, E, w), K = \{t_1, \ldots, t_k\})$

1: Set $\delta = \frac{1}{20 \ln k}$ and $p = \frac{1}{5}$.
2: Set $V_\perp \leftarrow V \setminus K$.
3: for $j$ from 1 to $k$ do
4:  Choose independently at random $g_j$ distributed according to $\text{Geo}(p)$.
5:  Set $R_j \leftarrow (1 + \delta)^{g_j}$.
6:  Set $V_j \leftarrow \text{Create-Cluster}(G, V_\perp, t_j, R_j)$.
7:  Remove all the vertices in $V_j$ from $V_\perp$.
8: end for
9: return the terminal-centered minor $M$ of $G$ induced by $V_1, \ldots, V_k$. 
## Noisy Voronoi

**Algorithm 2** $M = \text{Noisy-Voronoi}(G = (V, E, w), K = \{t_1, \ldots, t_k\})$

1. Set $\delta = \frac{1}{20 \ln k}$ and $p = \frac{1}{5}$.
2. Set $V_\perp \leftarrow V \setminus K$.
3. **for** $j$ from 1 to $k$ **do**
   4. Choose independently at random $g_j$ distributed according to $\text{Geo}(p)$.
   5. Set $R_j \leftarrow (1 + \delta)^{g_j}$.
   6. Set $V_j \leftarrow \text{Create-Cluster}(G, V_\perp, t_j, R_j)$.
   7. Remove all the vertices in $V_j$ from $V_\perp$.
4. **end for**
5. **return** the terminal-centered minor $M$ of $G$ induced by $V_1, \ldots, V_k$. 