# Graph Sparsification while Maintaining Cuts

Ramesh Hariharan
Strand Life Sciences

8 May 2011

## The Setting

- Graph $G$ with $n$ nodes and $m$ edges.
- Unweighted for this talk (weighted cases work similarly).
- $m >> n \log n$
- Obtain $G'$ with fewer edges but with all cuts of $G$ preserved approximately.
- $G'$ will be weighted.

## The Setting

- Graph $G$ with $n$ nodes and $m$ edges.
- Unweighted for this talk (weighted cases work similarly).
- $m >> n \log n$
- Obtain $G'$ with fewer edges but with all cuts of $G$ preserved approximately.
- $G'$ will be weighted.

## The Setting

- Graph $G$ with $n$ nodes and $m$ edges.
- Unweighted for this talk (weighted cases work similarly).
- $m >> n \log n$
- Obtain $G'$ with fewer edges but with all cuts of $G$ preserved approximately.
- $G'$ will be weighted.

## The Setting

- Graph $G$ with $n$ nodes and $m$ edges.
- Unweighted for this talk (weighted cases work similarly).
- $m >> n \log n$
- Obtain $G'$ with fewer edges but with all cuts of $G$ preserved approximately.
- $G'$ will be weighted.

## The Setting

- Graph $G$ with $n$ nodes and $m$ edges.
- Unweighted for this talk (weighted cases work similarly).
- $m >> n \log n$
- Obtain $G'$ with fewer edges but with all cuts of $G$ preserved approximately.
- $G'$ will be weighted.

## An Example

- Two nodes with $m$ edges connected the two.
- Replace by a single edge of weight $m$.
- The general case is more complex because there are many cuts in a graph.

## An Example

- Two nodes with $m$ edges connected the two.
- Replace by a single edge of weight $m$.
- The general case is more complex because there are many cuts in a graph.

## An Example

- Two nodes with $m$ edges connected the two.
- Replace by a single edge of weight $m$.
- The general case is more complex because there are many cuts in a graph.

## A Randomized Approach: Benczur, Karger

- Sample each edge $e$ with probability $p_e$ and give it weight $1/p_e$.
- For any cut, its expected weight in the new graph $G'$ equals its weight in $G$.
- Do ALL cuts in $G$ have weight in $G'$ that is $(1 \pm \epsilon)$ of the corresponding weight in $G$, w.h.p?
- And how many edges does $G'$ have?

## A Randomized Approach: Benczur, Karger

- Sample each edge $e$ with probability $p_e$ and give it weight $1/p_e$.
- For any cut, its expected weight in the new graph $G'$ equals its weight in $G$.
- Do ALL cuts in $G$ have weight in $G'$ that is $(1 \pm \epsilon)$ of the corresponding weight in $G$, w.h.p?
- And how many edges does $G'$ have?

## A Randomized Approach: Benczur, Karger

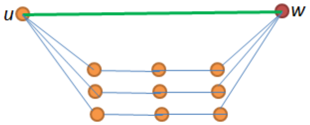
- Sample each edge $e$ with probability $p_e$ and give it weight $1/p_e$.
- For any cut, its expected weight in the new graph $G'$ equals its weight in $G$.
- Do ALL cuts in $G$ have weight in $G'$ that is $(1 \pm \epsilon)$ of the corresponding weight in $G$, w.h.p?
- And how many edges does $G'$ have?

## A Randomized Approach: Benczur, Karger

- Sample each edge $e$ with probability $p_e$ and give it weight $1/p_e$.
- For any cut, its expected weight in the new graph $G'$ equals its weight in $G$.
- Do ALL cuts in $G$ have weight in $G'$ that is $(1 \pm \epsilon)$ of the corresponding weight in $G$, w.h.p?
- And how many edges does $G'$ have?

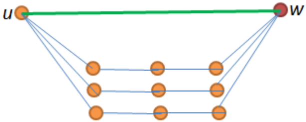
## What should $p_e$ be?

- $p_e \sim \frac{1}{d_e}$? ($d_e$ is min of the degrees of $e$'s endpoints). NO!
- $p_e \sim \frac{1}{k_e} \geq \frac{1}{d_e}$? ($k_e$ is the connectivity of $e$). NO!
- $p_e \sim \frac{\log n}{\epsilon^2} \frac{1}{k_e}$? MAYBE!



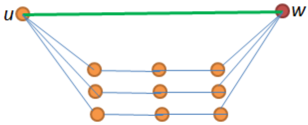
## What should $p_e$ be?

- $p_e \sim \frac{1}{d_e}$? ($d_e$ is min of the degrees of $e$'s endpoints). NO!
- $p_e \sim \frac{1}{k_e} \geq \frac{1}{d_e}$? ($k_e$ is the connectivity of $e$). NO!
- $p_e \sim \frac{\log n}{\epsilon^2} \frac{1}{k_e}$? MAYBE!



## What should $p_e$ be?

- $p_e \sim \frac{1}{d_e}$? ($d_e$ is min of the degrees of $e$'s endpoints). NO!
- $p_e \sim \frac{1}{k_e} \geq \frac{1}{d_e}$? ($k_e$ is the connectivity of $e$). NO!
- $p_e \sim \frac{\log n}{\epsilon^2}\frac{1}{k_e}$? MAYBE!



## Chernoff Bound for Sampled Cut Weight.

- Focus on a particular cut of size $\Delta$.
- Group edges in this cut into doubling value categories based on sampling probability.
- Consider one group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2}\frac{1}{2^i}$.
- For any $\Delta' \geq |\mathcal{S}|$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta') \leq e^{-\Theta(\epsilon^2 \frac{\log n}{\epsilon^2 2^i}\Delta')} = n^{-\Theta(\frac{\Delta'}{2^i})}$$

.

- We need $\epsilon\Delta'$ to add up at most $\epsilon\Delta$ over all groups.
- And we need $n^{-\Theta(\frac{\Delta'}{2^i})}$ to be small enough to offset the number of such groups $\mathcal{S}$ over all cuts.

## Chernoff Bound for Sampled Cut Weight.

- Focus on a particular cut of size $\Delta$.
- Group edges in this cut into doubling value categories based on sampling probability.
- Consider one group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2}\frac{1}{2^i}$.
- For any $\Delta' \geq |\mathcal{S}|$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta') \leq e^{-\Theta(\epsilon^2 \frac{\log n}{\epsilon^2 2^i}\Delta')} = n^{-\Theta(\frac{\Delta'}{2^i})}$$

.

- We need $\epsilon\Delta'$ to add up at most $\epsilon\Delta$ over all groups.
- And we need $n^{-\Theta(\frac{\Delta'}{2^i})}$ to be small enough to offset the number of such groups $\mathcal{S}$ over all cuts.

## Chernoff Bound for Sampled Cut Weight.

- Focus on a particular cut of size $\Delta$.
- Group edges in this cut into doubling value categories based on sampling probability.
- Consider one group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2}\frac{1}{2^i}$.
- For any $\Delta' \geq |\mathcal{S}|$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta') \leq e^{-\Theta(\epsilon^2 \frac{\log n}{\epsilon^2 2^i}\Delta')} = n^{-\Theta(\frac{\Delta'}{2^i})}$$

.

- We need $\epsilon\Delta'$ to add up at most $\epsilon\Delta$ over all groups.
- And we need $n^{-\Theta(\frac{\Delta'}{2^i})}$ to be small enough to offset the number of such groups $\mathcal{S}$ over all cuts.

## Chernoff Bound for Sampled Cut Weight.

- Focus on a particular cut of size $\Delta$.
- Group edges in this cut into doubling value categories based on sampling probability.
- Consider one group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2}\frac{1}{2^i}$.
- For any $\Delta' \geq |\mathcal{S}|$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta') \leq e^{-\Theta(\epsilon^2 \frac{\log n}{\epsilon^2 2^i}\Delta')} = n^{-\Theta(\frac{\Delta'}{2^i})}$$

.

- We need $\epsilon\Delta'$ to add up at most $\epsilon\Delta$ over all groups.
- And we need $n^{-\Theta(\frac{\Delta'}{2^i})}$ to be small enough to offset the number of such groups $\mathcal{S}$ over all cuts.

# Chernoff Bound for Sampled Cut Weight.

- Focus on a particular cut of size $\Delta$.
- Group edges in this cut into doubling value categories based on sampling probability.
- Consider one group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2}\frac{1}{2^i}$.
- For any $\Delta' \geq |\mathcal{S}|$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta') \leq e^{-\Theta(\epsilon^2 \frac{\log n}{\epsilon^2 2^i}\Delta')} = n^{-\Theta(\frac{\Delta'}{2^i})}$$

.

- We need $\epsilon\Delta'$ to add up at most $\epsilon\Delta$ over all groups.
- And we need $n^{-\Theta(\frac{\Delta'}{2^i})}$ to be small enough to offset the number of such groups $\mathcal{S}$ over all cuts.

# Chernoff Bound for Sampled Cut Weight.

- Focus on a particular cut of size $\Delta$.
- Group edges in this cut into doubling value categories based on sampling probability.
- Consider one group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2}\frac{1}{2^i}$.
- For any $\Delta' \geq |\mathcal{S}|$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta') \leq e^{-\Theta(\epsilon^2 \frac{\log n}{\epsilon^2 2^i}\Delta')} = n^{-\Theta(\frac{\Delta'}{2^i})}$$

.

- We need $\epsilon\Delta'$ to add up at most $\epsilon\Delta$ over all groups.
- And we need $n^{-\Theta(\frac{\Delta'}{2^i})}$ to be small enough to offset the number of such groups $\mathcal{S}$ over all cuts.

## Chernoff Bound for Sampled Cut Weight.

- Focus on a particular cut of size $\Delta$.
- Group edges in this cut into doubling value categories based on sampling probability.
- Consider one group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2}\frac{1}{2^i}$.
- For any $\Delta' \geq |\mathcal{S}|$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta') \leq e^{-\Theta(\epsilon^2 \frac{\log n}{\epsilon^2 2^i}\Delta')} = n^{-\Theta(\frac{\Delta'}{2^i})}$$

.

- We need $\epsilon\Delta'$ to add up at most $\epsilon\Delta$ over all groups.
- And we need $n^{-\Theta(\frac{\Delta'}{2^i})}$ to be small enough to offset the number of such groups $\mathcal{S}$ over all cuts.

## How many cuts?

- Focus on a particular cut of size $\Delta$, and the subset of edges with connectivity $\sim 2^i$.
- How many such distinct sets of edges exist, over all cuts?
- $n^{O(\Delta/2^i)}$! Will show later.

## How many cuts?

- Focus on a particular cut of size $\Delta$, and the subset of edges with connectivity $\sim 2^i$.
- How many such distinct sets of edges exist, over all cuts?
- $n^{O(\Delta/2^i)}$! Will show later.

## How many cuts?

- Focus on a particular cut of size $\Delta$, and the subset of edges with connectivity $\sim 2^i$.
- How many such distinct sets of edges exist, over all cuts?
- $n^{O(\Delta/2^i)}$! Will show later.

## How many cuts?

- Focus on a particular cut of size $\Delta$, and the subset of edges with connectivity $\sim 2^i$.
- How many such distinct sets of edges exist, over all cuts?
- $n^{O(\Delta/2^i)}$! Will show later.

# Putting it Together

- Focus on a particular cut of size $\Delta$, and the group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2 2^i}$.
- For this group,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}| \geq \epsilon\Delta) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$

.
- The number of distinct groups of edges $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$ (to be shown).
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon\Delta$.
- There are at most $\log n$ groups in each cut.
- So every cut has deviation at most $\epsilon\Delta \log n$. But we need $\Delta$!

## Putting it Together

- Focus on a particular cut of size $\Delta$, and the group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2 2^i}$.
- For this group,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}| \geq \epsilon\Delta) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$

.
- The number of distinct groups of edges $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$ (to be shown).
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon\Delta$.
- There are at most $\log n$ groups in each cut.
- So every cut has deviation at most $\epsilon\Delta \log n$. But we need $\Delta$!

## Putting it Together

- Focus on a particular cut of size $\Delta$, and the group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2 2^i}$.
- For this group,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}| \geq \epsilon\Delta) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$

.

- The number of distinct groups of edges $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$ (to be shown).
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon\Delta$.
- There are at most $\log n$ groups in each cut.
- So every cut has deviation at most $\epsilon\Delta \log n$. But we need $\Delta$!

## Putting it Together

- Focus on a particular cut of size $\Delta$, and the group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2 2^i}$.
- For this group,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}| \geq \epsilon\Delta) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$

.

- The number of distinct groups of edges $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$ (to be shown).
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon\Delta$.
- There are at most $\log n$ groups in each cut.
- So every cut has deviation at most $\epsilon\Delta \log n$. But we need $\Delta$!

## Putting it Together

- Focus on a particular cut of size $\Delta$, and the group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2 2^i}$.
- For this group,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}| \geq \epsilon\Delta) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$

.

- The number of distinct groups of edges $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$ (to be shown).
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon\Delta$.
- There are at most $\log n$ groups in each cut.
- So every cut has deviation at most $\epsilon\Delta \log n$. But we need $\Delta$!

## Putting it Together

- Focus on a particular cut of size $\Delta$, and the group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2 2^i}$.
- For this group,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}| \geq \epsilon\Delta) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$

  .
- The number of distinct groups of edges $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$ (to be shown).
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon\Delta$.
- There are at most $\log n$ groups in each cut.
- So every cut has deviation at most $\epsilon\Delta \log n$. But we need $\Delta$!

## Putting it Together

- Focus on a particular cut of size $\Delta$, and the group $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log n}{\epsilon^2 2^i}$.
- For this group,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}| \geq \epsilon\Delta) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$

.

- The number of distinct groups of edges $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$ (to be shown).
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon\Delta$.
- There are at most $\log n$ groups in each cut.
- So every cut has deviation at most $\epsilon\Delta \log n$. But we need $\Delta$!

## Another Attempt

- Up sampling probabilities to $\sim \frac{\log^2 n}{\epsilon^2 k_e}$.
- Focus on a particular cut of size $\Delta$, and the set $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log^2 n}{\epsilon^2 * 2^i}$.
- $$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon \frac{\Delta}{\log n}) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$
  .
- The number of such groups $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon \frac{\Delta}{\log n}$.
- So every cut has deviation at most $\epsilon\Delta$ over all $\log n$ groups.

## Another Attempt

- Up sampling probabilities to $\sim \frac{\log^2 n}{\epsilon^2 k_e}$.
- Focus on a particular cut of size $\Delta$, and the set $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log^2 n}{\epsilon^2 * 2^i}$.
-
$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon \frac{\Delta}{\log n}) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$
.
- The number of such groups $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon \frac{\Delta}{\log n}$.
- So every cut has deviation at most $\epsilon\Delta$ over all $\log n$ groups.

## Another Attempt

- Up sampling probabilities to $\sim \frac{\log^2 n}{\epsilon^2 k_e}$.
- Focus on a particular cut of size $\Delta$, and the set $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log^2 n}{\epsilon^2 * 2^i}$.
- $$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon \frac{\Delta}{\log n}) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$
  .
- The number of such groups $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon \frac{\Delta}{\log n}$.
- So every cut has deviation at most $\epsilon\Delta$ over all $\log n$ groups.

## Another Attempt

- Up sampling probabilities to $\sim \frac{\log^2 n}{\epsilon^2 k_e}$.
- Focus on a particular cut of size $\Delta$, and the set $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log^2 n}{\epsilon^2 * 2^i}$.
- $$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon \frac{\Delta}{\log n}) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$
  .
- The number of such groups $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon \frac{\Delta}{\log n}$.
- So every cut has deviation at most $\epsilon\Delta$ over all $\log n$ groups.

## Another Attempt

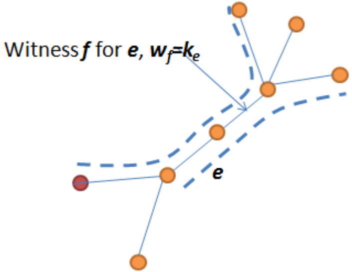
- Up sampling probabilities to $\sim \frac{\log^2 n}{\epsilon^2 k_e}$.
- Focus on a particular cut of size $\Delta$, and the set $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log^2 n}{\epsilon^2 * 2^i}$.
- $$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon \frac{\Delta}{\log n}) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$
  .
- The number of such groups $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon \frac{\Delta}{\log n}$.
- So every cut has deviation at most $\epsilon\Delta$ over all $\log n$ groups.

## Another Attempt

- Up sampling probabilities to $\sim \frac{\log^2 n}{\epsilon^2 k_e}$.
- Focus on a particular cut of size $\Delta$, and the set $\mathcal{S}$ of edges with sampling probabilities $\sim \frac{\log^2 n}{\epsilon^2 * 2^i}$.
- $$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon \frac{\Delta}{\log n}) \leq n^{-\Theta(\frac{\Delta}{2^i})}$$
  .
- The number of such groups $\mathcal{S}$ over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.
- So in every cut of size $\Delta$, the corresponding group contributes a deviation of $\epsilon \frac{\Delta}{\log n}$.
- So every cut has deviation at most $\epsilon\Delta$ over all $\log n$ groups.

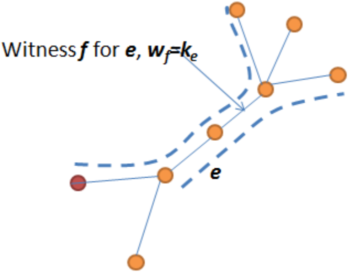
## Expected Size of Sampled Graph

- We show that $\sum_e \frac{1}{k_e} \leq n-1$. So the expected number of edges in the sampled graph is $\leq \frac{\log^2 n}{\epsilon^2}(n-1)$.
- Consider the Gomory-Hu (GH) tree. Each Gomory-Hu edge $f$ has weight $w_f$ equal to the number of graph edges that cross it.
- $e$ crosses a witness Gomory-Hu edge with weight $k_e$.
- $\sum \frac{1}{k_e} \leq \sum_f w_f * 1/w_f = n-1$.



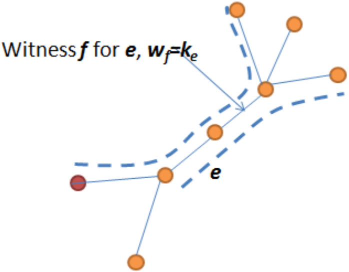
## Expected Size of Sampled Graph

- We show that $\sum_e \frac{1}{k_e} \leq n-1$. So the expected number of edges in the sampled graph is $\leq \frac{\log^2 n}{\epsilon^2}(n-1)$.
- Consider the Gomory-Hu (GH) tree. Each Gomory-Hu edge $f$ has weight $w_f$ equal to the number of graph edges that cross it.
- $e$ crosses a witness Gomory-Hu edge with weight $k_e$.
- $\sum \frac{1}{k_e} \leq \sum_f w_f * 1/w_f = n-1$.



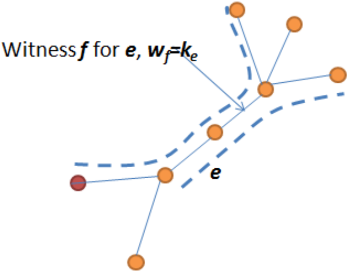
## Expected Size of Sampled Graph

- We show that $\sum_e \frac{1}{k_e} \leq n - 1$. So the expected number of edges in the sampled graph is $\leq \frac{\log^2 n}{\epsilon^2}(n-1)$.
- Consider the Gomory-Hu (GH) tree. Each Gomory-Hu edge $f$ has weight $w_f$ equal to the number of graph edges that cross it.
- $e$ crosses a witness Gomory-Hu edge with weight $k_e$.
- $\sum \frac{1}{k_e} \leq \sum_f w_f * 1/w_f = n - 1$.



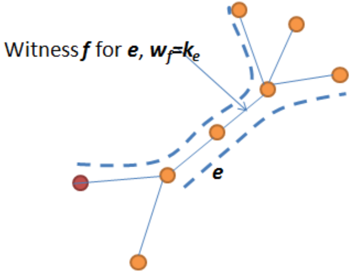
## Expected Size of Sampled Graph

- We show that $\sum_e \frac{1}{k_e} \leq n-1$. So the expected number of edges in the sampled graph is $\leq \frac{\log^2 n}{\epsilon^2}(n-1)$.
- Consider the Gomory-Hu (GH) tree. Each Gomory-Hu edge $f$ has weight $w_f$ equal to the number of graph edges that cross it.
- $e$ crosses a witness Gomory-Hu edge with weight $k_e$.
- $\sum \frac{1}{k_e} \leq \sum_f w_f * 1/w_f = n-1$.



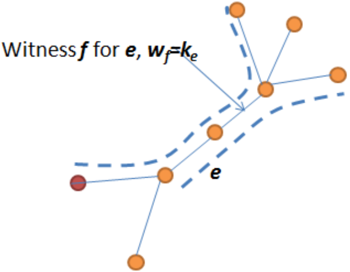
## Expected Size of Sampled Graph

- We show that $\sum_e \frac{1}{k_e} \leq n-1$. So the expected number of edges in the sampled graph is $\leq \frac{\log^2 n}{\epsilon^2}(n-1)$.
- Consider the Gomory-Hu (GH) tree. Each Gomory-Hu edge $f$ has weight $w_f$ equal to the number of graph edges that cross it.
- $e$ crosses a witness Gomory-Hu edge with weight $k_e$.
- $\sum \frac{1}{k_e} \leq \sum_f w_f * 1/w_f = n-1.$



## Expected Size of Sampled Graph

- We show that $\sum_e \frac{1}{k_e} \leq n - 1$. So the expected number of edges in the sampled graph is $\leq \frac{\log^2 n}{\epsilon^2}(n-1)$.
- Consider the Gomory-Hu (GH) tree. Each Gomory-Hu edge $f$ has weight $w_f$ equal to the number of graph edges that cross it.
- $e$ crosses a witness Gomory-Hu edge with weight $k_e$.
- $\sum \frac{1}{k_e} \leq \sum_f w_f * 1/w_f = n - 1$.



## Wrapping Up: Sampling by Connectivity

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 k_e}$ ($k_e$ is the connectivity of $e$).
- The expected number of edges in the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.
- And each cut is preserved within a $(1 \pm \epsilon)$ multiplicative factor, with inverse polynomial failure probability.

## Wrapping Up: Sampling by Connectivity

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 k_e}$ ($k_e$ is the connectivity of $e$).
- The expected number of edges in the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.
- And each cut is preserved within a $(1 \pm \epsilon)$ multiplicative factor, with inverse polynomial failure probability.

## Wrapping Up: Sampling by Connectivity

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 k_e}$ ($k_e$ is the connectivity of $e$).
- The expected number of edges in the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.
- And each cut is preserved within a $(1 \pm \epsilon)$ multiplicative factor, with inverse polynomial failure probability.

## Wrapping Up: Sampling by Connectivity

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 k_e}$ ($k_e$ is the connectivity of $e$).
- The expected number of edges in the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.
- And each cut is preserved within a $(1 \pm \epsilon)$ multiplicative factor, with inverse polynomial failure probability.

## Wrapping Up: Sampling by Connectivity

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 k_e}$ ($k_e$ is the connectivity of $e$).
- The expected number of edges in the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.
- And each cut is preserved within a $(1 \pm \epsilon)$ multiplicative factor, with inverse polynomial failure probability.

## Counting Cuts

- Consider a cut of size $\Delta$.
- Define its $2^i$-projection to be the subset of edges with connectivity $\sim 2^i$.
- How many distinct $2^i$-projections exist over all cuts of size $\Delta$? We show $n^{O(\Delta/2^i)}$.

## Counting Cuts

- Consider a cut of size $\Delta$.
- Define its $2^i$-projection to be the subset of edges with connectivity $\sim 2^i$.
- How many distinct $2^i$-projections exist over all cuts of size $\Delta$? We show $n^{O(\Delta/2^i)}$.

## Counting Cuts

- Consider a cut of size $\Delta$.
- Define its $2^i$-projection to be the subset of edges with connectivity $\sim 2^i$.
- How many distinct $2^i$-projections exist over all cuts of size $\Delta$? We show $n^{O(\Delta/2^i)}$.

## Counting Cuts

- Consider a cut of size $\Delta$.
- Define its $2^i$-projection to be the subset of edges with connectivity $\sim 2^i$.
- How many distinct $2^i$-projections exist over all cuts of size $\Delta$? We show $n^{O(\Delta/2^i)}$.

## Counting Cuts

- Consider a cut of size $\Delta$.
- Define its $2^i$-projection to be the subset of edges with connectivity $\sim 2^i$.
- How many distinct $2^i$-projections exist over all cuts of size $\Delta$? We show $n^{O(\Delta/2^i)}$.

## Recall Karger's Cut Counting Method

- Randomly choose edges and compress.
- Let $k$ be the min-cut size.
- Probability of being left with a particular cut of size $\Delta$ is

$$\geq (1 - \frac{\Delta}{nk/2})(1 - \frac{\Delta}{(n-1)k/2}) \cdots (1 - \frac{\Delta}{(\frac{2\Delta}{k}+1)k/2})$$

$$\geq (\frac{n - 2\Delta/k}{n})(\frac{n-1-2\Delta/k}{n-1}) \cdots (\frac{n-(n-\frac{2\Delta}{k}-1)-2\Delta/k}{\frac{2\Delta}{k}+1})$$

$$\geq n^{\frac{-2\Delta}{k}}$$

- So the number of distinct cuts of size $\Delta$ in a graph with min-cut $k$ is at most $n^{\frac{2\Delta}{k}}$.

## Recall Karger's Cut Counting Method

- Randomly choose edges and compress.
- Let $k$ be the min-cut size.
- Probability of being left with a particular cut of size $\Delta$ is

$$\geq (1-\frac{\Delta}{nk/2})(1-\frac{\Delta}{(n-1)k/2})\cdots(1-\frac{\Delta}{(\frac{2\Delta}{k}+1)k/2})$$

$$\geq (\frac{n-2\Delta/k}{n})(\frac{n-1-2\Delta/k}{n-1})\cdots(\frac{n-(n-\frac{2\Delta}{k}-1)-2\Delta/k}{\frac{2\Delta}{k}+1})$$

$$\geq n^{\frac{-2\Delta}{k}}$$

- So the number of distinct cuts of size $\Delta$ in a graph with min-cut $k$ is at most $n^{\frac{2\Delta}{k}}$.

## Recall Karger's Cut Counting Method

- Randomly choose edges and compress.
- Let $k$ be the min-cut size.
- Probability of being left with a particular cut of size $\Delta$ is

$$\geq (1 - \frac{\Delta}{nk/2})(1 - \frac{\Delta}{(n-1)k/2}) \cdots (1 - \frac{\Delta}{(\frac{2\Delta}{k}+1)k/2})$$

$$\geq (\frac{n - 2\Delta/k}{n})(\frac{n-1-2\Delta/k}{n-1}) \cdots (\frac{n-(n-\frac{2\Delta}{k}-1)-2\Delta/k}{\frac{2\Delta}{k}+1})$$

$$\geq n^{\frac{-2\Delta}{k}}$$

- So the number of distinct cuts of size $\Delta$ in a graph with min-cut $k$ is at most $n^{\frac{2\Delta}{k}}$.

## Recall Karger's Cut Counting Method

- Randomly choose edges and compress.
- Let $k$ be the min-cut size.
- Probability of being left with a particular cut of size $\Delta$ is

$$\geq (1 - \frac{\Delta}{nk/2})(1 - \frac{\Delta}{(n-1)k/2}) \cdots (1 - \frac{\Delta}{(\frac{2\Delta}{k}+1)k/2})$$

$$\geq (\frac{n - 2\Delta/k}{n})(\frac{n-1-2\Delta/k}{n-1}) \cdots (\frac{n-(n-\frac{2\Delta}{k}-1)-2\Delta/k}{\frac{2\Delta}{k}+1})$$

$$\geq n^{\frac{-2\Delta}{k}}$$

- So the number of distinct cuts of size $\Delta$ in a graph with min-cut $k$ is at most $n^{\frac{2\Delta}{k}}$.

## Recall Karger's Cut Counting Method

- Randomly choose edges and compress.
- Let $k$ be the min-cut size.
- Probability of being left with a particular cut of size $\Delta$ is

$$\geq (1 - \frac{\Delta}{nk/2})(1 - \frac{\Delta}{(n-1)k/2}) \cdots (1 - \frac{\Delta}{(\frac{2\Delta}{k} + 1)k/2})$$

$$\geq (\frac{n - 2\Delta/k}{n})(\frac{n - 1 - 2\Delta/k}{n-1}) \cdots (\frac{n - (n - \frac{2\Delta}{k} - 1) - 2\Delta/k}{\frac{2\Delta}{k} + 1})$$

$$\geq n^{\frac{-2\Delta}{k}}$$

- So the number of distinct cuts of size $\Delta$ in a graph with min-cut $k$ is at most $n^{\frac{2\Delta}{k}}$.

## Counting $2^i$-projections

- Randomly choose edges and compress.
- If min-cut was $2^i$ then done.
- What if there are vertices with degree $< 2^i$?
- Edges incident on such vertices are not part of a $2^i$-projection.
- So **split-off** these vertices.

## Counting $2^i$-projections

- Randomly choose edges and compress.
- If min-cut was $2^i$ then done.
- What if there are vertices with degree $< 2^i$?
- Edges incident on such vertices are not part of a $2^i$-projection.
- So **split-off** these vertices.

## Counting $2^i$-projections

- Randomly choose edges and compress.
- If min-cut was $2^i$ then done.
- What if there are vertices with degree $< 2^i$?
- Edges incident on such vertices are not part of a $2^i$-projection.
- So **split-off** these vertices.

## Counting $2^i$-projections

- Randomly choose edges and compress.
- If min-cut was $2^i$ then done.
- What if there are vertices with degree $< 2^i$?
- Edges incident on such vertices are not part of a $2^i$-projection.
- So **split-off** these vertices.

## Counting $2^i$-projections

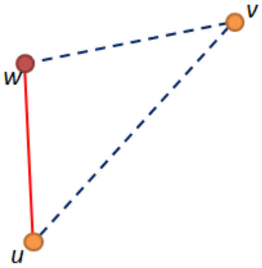
- Randomly choose edges and compress.
- If min-cut was $2^i$ then done.
- What if there are vertices with degree $< 2^i$?
- Edges incident on such vertices are not part of a $2^i$-projection.
- So **split-off** these vertices.

## Counting $2^i$-projections

- Randomly choose edges and compress.
- If min-cut was $2^i$ then done.
- What if there are vertices with degree $< 2^i$?
- Edges incident on such vertices are not part of a $2^i$-projection.
- So **split-off** these vertices.

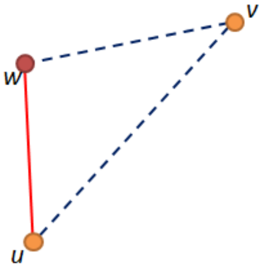
## Splitting Off

- Edges incident on a vertex $v$ can be paired and 'shortcut'.
- So $v$ gets removed from the graph.
- The connectivity of edges with connectivity $>= 2^i$ does not fall below $2^i$.
- And no cut increases in size (to see this, note that any edge across a cut after splitting-off must have a sub-edge across the cut before splitting-off), so a cut of size $\Delta$ remains of size at most $\Delta$.



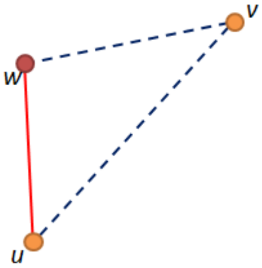
## Splitting Off

- Edges incident on a vertex $v$ can be paired and 'shortcut'.
- So $v$ gets removed from the graph.
- The connectivity of edges with connectivity $>= 2^i$ does not fall below $2^i$.
- And no cut increases in size (to see this, note that any edge across a cut after splitting-off must have a sub-edge across the cut before splitting-off), so a cut of size $\Delta$ remains of size at most $\Delta$.



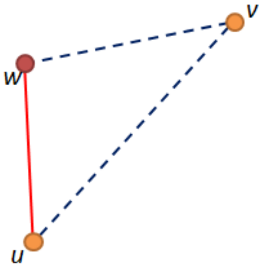
## Splitting Off

- Edges incident on a vertex $v$ can be paired and 'shortcut'.
- So $v$ gets removed from the graph.
- The connectivity of edges with connectivity $>= 2^i$ does not fall below $2^i$.
- And no cut increases in size (to see this, note that any edge across a cut after splitting-off must have a sub-edge across the cut before splitting-off), so a cut of size $\Delta$ remains of size at most $\Delta$.



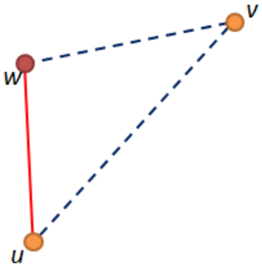
## Splitting Off

- Edges incident on a vertex $v$ can be paired and 'shortcut'.
- So $v$ gets removed from the graph.
- The connectivity of edges with connectivity $>= 2^i$ does not fall below $2^i$.
- And no cut increases in size (to see this, note that any edge across a cut after splitting-off must have a sub-edge across the cut before splitting-off), so a cut of size $\Delta$ remains of size at most $\Delta$.



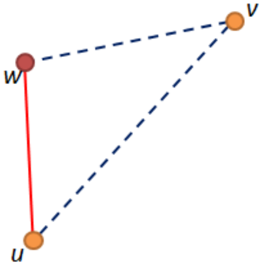
## Splitting Off

- Edges incident on a vertex $v$ can be paired and 'shortcut'.
- So $v$ gets removed from the graph.
- The connectivity of edges with connectivity $>= 2^i$ does not fall below $2^i$.
- And no cut increases in size (to see this, note that any edge across a cut after splitting-off must have a sub-edge across the cut before splitting-off), so a cut of size $\Delta$ remains of size at most $\Delta$.



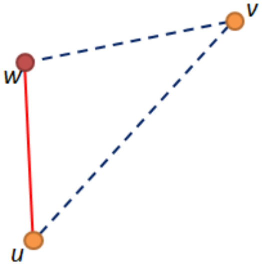
## Why Splitting Off?

- Compressing an edge causes potential increase in the cut size $\Delta$.
- Deleting an edge causes potential decrease in connectivity $2^i$.
- Adding an edge causes potential increase in the cut size $\Delta$.
- Only splitting ensures that connnectivity $2^i$ does not drop and cut size $\Delta$ does not increase.



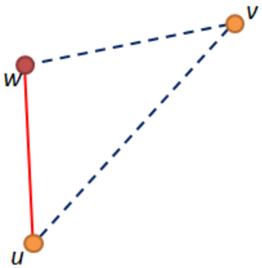
## Why Splitting Off?

- Compressing an edge causes potential increase in the cut size $\Delta$.
- Deleting an edge causes potential decrease in connectivity $2^i$.
- Adding an edge causes potential increase in the cut size $\Delta$.
- Only splitting ensures that connnectivity $2^i$ does not drop and cut size $\Delta$ does not increase.



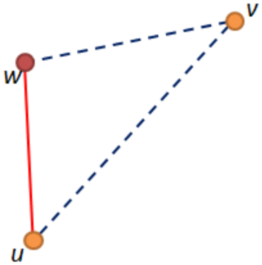
## Why Splitting Off?

- Compressing an edge causes potential increase in the cut size $\Delta$.
- Deleting an edge causes potential decrease in connectivity $2^i$.
- Adding an edge causes potential increase in the cut size $\Delta$.
- Only splitting ensures that connnectivity $2^i$ does not drop and cut size $\Delta$ does not increase.



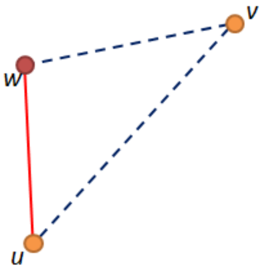
## Why Splitting Off?

- Compressing an edge causes potential increase in the cut size $\Delta$.
- Deleting an edge causes potential decrease in connectivity $2^i$.
- Adding an edge causes potential increase in the cut size $\Delta$.
- Only splitting ensures that connnectivity $2^i$ does not drop and cut size $\Delta$ does not increase.



## Why Splitting Off?

- Compressing an edge causes potential increase in the cut size $\Delta$.
- Deleting an edge causes potential decrease in connectivity $2^i$.
- Adding an edge causes potential increase in the cut size $\Delta$.
- Only splitting ensures that connnectivity $2^i$ does not drop and cut size $\Delta$ does not increase.



## Back to counting $2^i$-projections

- Randomly choose edges and compress.
- If a vertex with degree $< 2^i$ is created, split it off.
- Edges in any $2^i$-projection stay $2^i$ connected.
- So such edges stay intact even as vertices are split off.
- Probability of being left with a particular $2^i$-projection (of cuts of size $\Delta$) is

$$\geq (1 - \Delta/n2^i)(1 - \Delta/(n-1)2^i) \cdots \geq n^{-\Delta/2^i}$$

- The number of distinct $2^i$-projections over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.

## Back to counting $2^i$-projections

- Randomly choose edges and compress.
- If a vertex with degree $< 2^i$ is created, split it off.
- Edges in any $2^i$-projection stay $2^i$ connected.
- So such edges stay intact even as vertices are split off.
- Probability of being left with a particular $2^i$-projection (of cuts of size $\Delta$) is

$$\geq (1 - \Delta/n2^i)(1 - \Delta/(n-1)2^i) \cdots \geq n^{-\Delta/2^i}$$

- The number of distinct $2^i$-projections over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.

## Back to counting $2^i$-projections

- Randomly choose edges and compress.
- If a vertex with degree $< 2^i$ is created, split it off.
- Edges in any $2^i$-projection stay $2^i$ connected.
- So such edges stay intact even as vertices are split off.
- Probability of being left with a particular $2^i$-projection (of cuts of size $\Delta$) is

$$\geq (1 - \Delta/n2^i)(1 - \Delta/(n-1)2^i) \cdots \geq n^{-\Delta/2^i}$$

- The number of distinct $2^i$-projections over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.

## Back to counting $2^i$-projections

- Randomly choose edges and compress.
- If a vertex with degree $< 2^i$ is created, split it off.
- Edges in any $2^i$-projection stay $2^i$ connected.
- So such edges stay intact even as vertices are split off.
- Probability of being left with a particular $2^i$-projection (of cuts of size $\Delta$) is

$$\geq (1 - \Delta/n2^i)(1 - \Delta/(n-1)2^i)\cdots \geq n^{-\Delta/2^i}$$

- The number of distinct $2^i$-projections over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.

## Back to counting $2^i$-projections

- Randomly choose edges and compress.
- If a vertex with degree $< 2^i$ is created, split it off.
- Edges in any $2^i$-projection stay $2^i$ connected.
- So such edges stay intact even as vertices are split off.
- Probability of being left with a particular $2^i$-projection (of cuts of size $\Delta$) is

$$\geq (1 - \Delta/n2^i)(1 - \Delta/(n-1)2^i) \cdots \geq n^{-\Delta/2^i}$$

- The number of distinct $2^i$-projections over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.

## Back to counting $2^i$-projections

- Randomly choose edges and compress.
- If a vertex with degree $< 2^i$ is created, split it off.
- Edges in any $2^i$-projection stay $2^i$ connected.
- So such edges stay intact even as vertices are split off.
- Probability of being left with a particular $2^i$-projection (of cuts of size $\Delta$) is

$$\geq (1 - \Delta/n2^i)(1 - \Delta/(n-1)2^i) \cdots \geq n^{-\Delta/2^i}$$

- The number of distinct $2^i$-projections over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.

## Back to counting $2^i$-projections

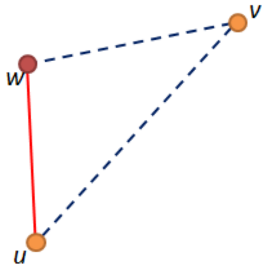
- Randomly choose edges and compress.
- If a vertex with degree $< 2^i$ is created, split it off.
- Edges in any $2^i$-projection stay $2^i$ connected.
- So such edges stay intact even as vertices are split off.
- Probability of being left with a particular $2^i$-projection (of cuts of size $\Delta$) is

$$\geq (1 - \Delta/n2^i)(1 - \Delta/(n-1)2^i) \cdots \geq n^{-\Delta/2^i}$$

- The number of distinct $2^i$-projections over all cuts of size $\Delta$ is $n^{O(\frac{\Delta}{2^i})}$.

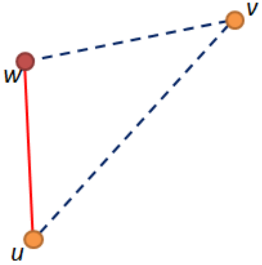
## Back to Splitting Off

- We need to prove that:
- Given edge *uv*, there exists *vw* such that:
- Removing *uv*, *vw* and adding *uw* ensures the following:
- All other edges with connectivity $\alpha = 2^i$ or higher remain at least $\alpha$ connected.



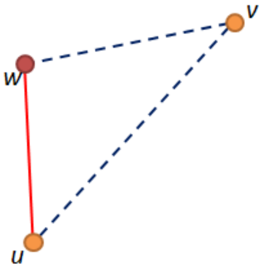
## Back to Splitting Off

- We need to prove that:
- Given edge $uv$, there exists $vw$ such that:
- Removing $uv$, $vw$ and adding $uw$ ensures the following:
- All other edges with connectivity $\alpha = 2^i$ or higher remain at least $\alpha$ connected.



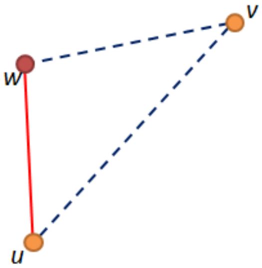
## Back to Splitting Off

- We need to prove that:
- Given edge *uv*, there exists *vw* such that:
- Removing *uv*, *vw* and adding *uw* ensures the following:
- All other edges with connectivity $\alpha = 2^i$ or higher remain at least $\alpha$ connected.



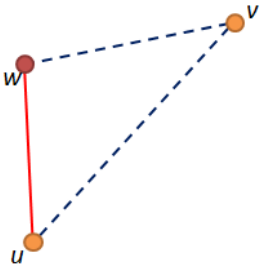
## Back to Splitting Off

- We need to prove that:
- Given edge *uv*, there exists *vw* such that:
- Removing *uv*, *vw* and adding *uw* ensures the following:
- All other edges with connectivity $\alpha = 2^i$ or higher remain at least $\alpha$ connected.



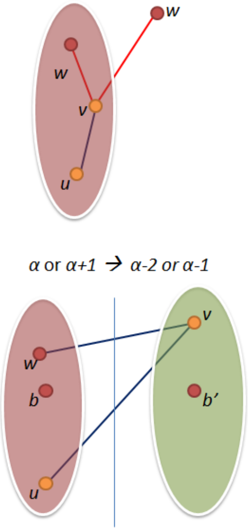
## Back to Splitting Off

- We need to prove that:
- Given edge *uv*, there exists *vw* such that:
- Removing *uv*, *vw* and adding *uw* ensures the following:
- All other edges with connectivity $\alpha = 2^i$ or higher remain at least $\alpha$ connected.



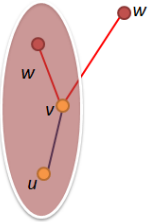
## The Challenge with Splitting Off

- The only cuts that reduce in size are those which split *u* and *v*.
- If there exists such a cut of size $\alpha$ or $\alpha + 1$, it will drop below $\alpha$ iff *w* is on the same size as *u* in this cut.
- A problem if such a cut splits a critical vertex pair $b, b'$ whose connectivity must be maintained at $\alpha$.

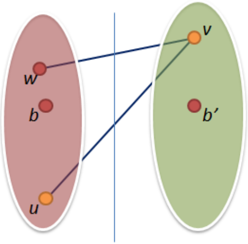


## The Challenge with Splitting Off

- The only cuts that reduce in size are those which split *u* and *v*.
- If there exists such a cut of size $\alpha$ or $\alpha + 1$, it will drop below $\alpha$ iff *w* is on the same size as *u* in this cut.
- A problem if such a cut splits a critical vertex pair $b, b'$ whose connectivity must be maintained at $\alpha$.

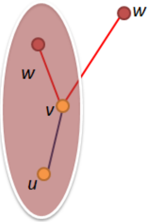


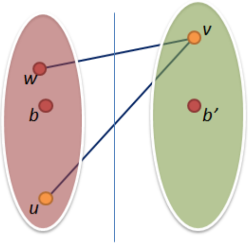

α or α+1 → α-2 or α-1



## The Challenge with Splitting Off

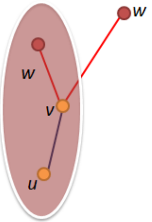
- The only cuts that reduce in size are those which split *u* and *v*.
- If there exists such a cut of size $\alpha$ or $\alpha + 1$, it will drop below $\alpha$ iff *w* is on the same size as *u* in this cut.
- A problem if such a cut splits a critical vertex pair $b, b'$ whose connectivity must be maintained at $\alpha$.

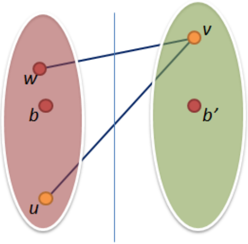


## The Challenge with Splitting Off

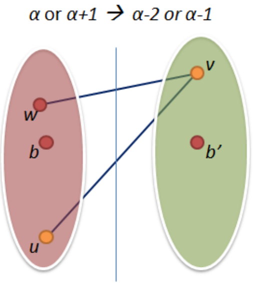
- The only cuts that reduce in size are those which split *u* and *v*.
- If there exists such a cut of size $\alpha$ or $\alpha + 1$, it will drop below $\alpha$ iff *w* is on the same size as *u* in this cut.
- A problem if such a cut splits a critical vertex pair *b*, *b*′ whose connectivity must be maintained at $\alpha$.



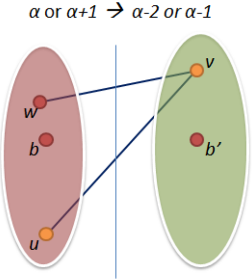
## Proof of Splitting Off

- $v$ must have a neighbour on the right side of the cut.
- Otherwise, move $v$ to the left and the cut size falls below $\alpha$.
- So $b$ and $b'$ are less than $\alpha$ connected, a contradiction.



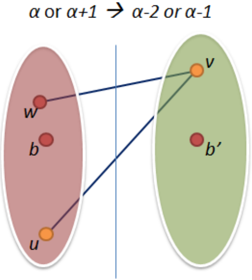
## Proof of Splitting Off

- $v$ must have a neighbour on the right side of the cut.
- Otherwise, move $v$ to the left and the cut size falls below $\alpha$.
- So $b$ and $b'$ are less than $\alpha$ connected, a contradiction.



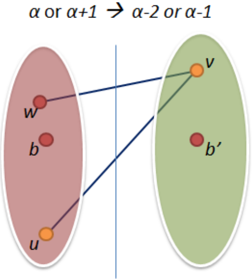
## Proof of Splitting Off

- $v$ must have a neighbour on the right side of the cut.
- Otherwise, move $v$ to the left and the cut size falls below $\alpha$.
- So $b$ and $b'$ are less than $\alpha$ connected, a contradiction.



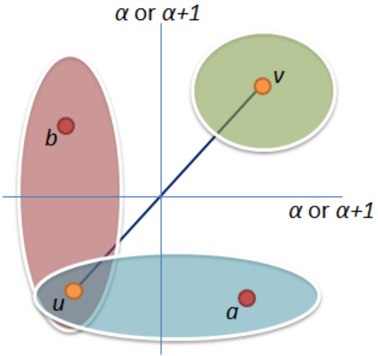
## Proof of Splitting Off

- $v$ must have a neighbour on the right side of the cut.
- Otherwise, move $v$ to the left and the cut size falls below $\alpha$.
- So $b$ and $b'$ are less than $\alpha$ connected, a contradiction.



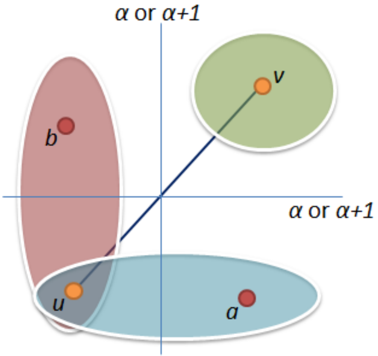
## Proof of Splitting Off Contd.

- But how do we find a $w$ that is on the same side as $v$ in all critical cuts?
- What if there are crossing critical cuts? .



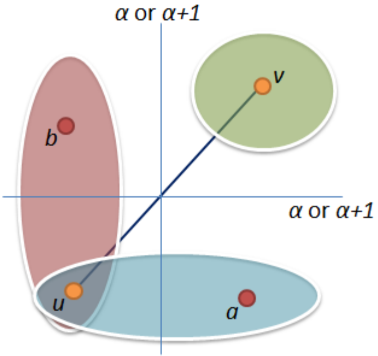
## Proof of Splitting Off Contd.

- But how do we find a $w$ that is on the same side as $v$ in all critical cuts?
- What if there are crossing critical cuts? .



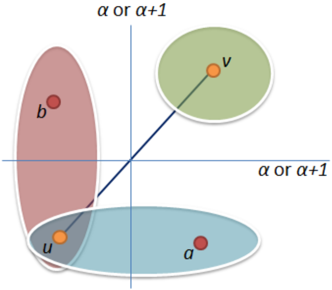
## Proof of Splitting Off Contd.

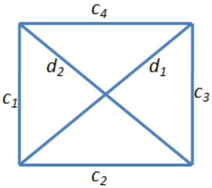
- But how do we find a $w$ that is on the same side as $v$ in all critical cuts?
- What if there are crossing critical cuts? .



## Critical Cuts do not cross.

- If all vertices have even degrees!




$c_1 + c_3 + d_1 + d_2 = \alpha$ or $\alpha+1$

$c_2 + c_4 + d_1 + d_2 = \alpha$ or $\alpha+1$

$c_1 + c_4 + d_2 >= \alpha$

$c_2 + c_3 + d_2 >= \alpha$

$d_1 >= 1$

$\Rightarrow d_1 = 1$

$c_1 + c_3 + d_2 = \alpha$

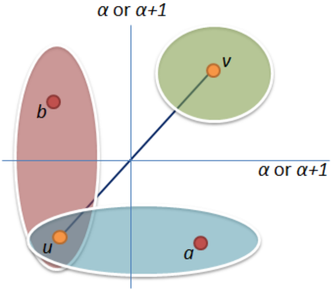
$c_1 + c_4 + d_2 = \alpha$

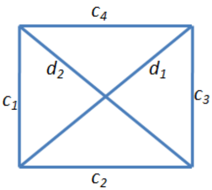
$\Rightarrow c_3 = c_4, d_1 = 1$

$\Rightarrow c_3 + c_4 + d_1$ is odd

## Critical Cuts do not cross.

- If all vertices have even degrees!




$c_1 + c_3 + d_1 + d_2 = \alpha$ or $\alpha+1$

$c_2 + c_4 + d_1 + d_2 = \alpha$ or $\alpha+1$

$c_1 + c_4 + d_2 >= \alpha$

$c_2 + c_3 + d_2 >= \alpha$

$d_1 >= 1$

$\Rightarrow d_1 = 1$

$c_1 + c_3 + d_2 = \alpha$

$c_1 + c_4 + d_2 = \alpha$

$\Rightarrow c_3 = c_4, d_1 = 1$

$\Rightarrow c_3 + c_4 + d_1$ is odd

## Splitting Off: Wrapping Up.

- How do we handle odd degrees?
- Simply double each edge! Cut sizes and connectivities double. Still good enough to estimate number of cuts.
- And splitting off and edge compression preserve evenness.

## Splitting Off: Wrapping Up.

- How do we handle odd degrees?
- Simply double each edge! Cut sizes and connectivities double. Still good enough to estimate number of cuts.
- And splitting off and edge compression preserve evenness.

## Splitting Off: Wrapping Up.

- How do we handle odd degrees?
- Simply double each edge! Cut sizes and connectivities double. Still good enough to estimate number of cuts.
- And splitting off and edge compression preserve evenness.

## Splitting Off: Wrapping Up.

- How do we handle odd degrees?
- Simply double each edge! Cut sizes and connectivities double. Still good enough to estimate number of cuts.
- And splitting off and edge compression preserve evenness.

## Computing Sampling Probabilities

- It suffices to underestimate edge connectivites, i.e., compute $k'_e \leq k_e$.
- Because sampling probabilities are used only in the Chernoff bound, which has the form:

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon x) \leq n^{-\Theta(\frac{x}{2^i})}$$

.

## Computing Sampling Probabilities

- It suffices to underestimate edge connectivites, i.e., compute $k'_e \leq k_e$.
- Because sampling probabilities are used only in the Chernoff bound, which has the form:

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon x) \leq n^{-\Theta(\frac{x}{2^i})}$$

.

## Computing Sampling Probabilities

- It suffices to underestimate edge connectivites, i.e., compute $k'_e \leq k_e$.
- Because sampling probabilities are used only in the Chernoff bound, which has the form:

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon x) \leq n^{-\Theta(\frac{x}{2^i})}$$

.

## Sampling using Nagamochi-Ibaraki Trees

- A collection of edge-disjoint forests.
- If $u$ and $v$ are connected in forest $i$, they are also connected in forests $1 \dots i-1$.
- If edge $e = uv$ is in tree $i$, then $i = k'_e \leq k_e$.
- So sampling with probability $\frac{\log^2 n}{\epsilon^2 k'_e}$ preserves all cuts within $1 \pm \epsilon$ w.h.p.
- $\sum \frac{1}{k'_e} \leq n \log n$ (as opposed to $\sum \frac{1}{k_e} \leq n$)
- Expected number of edges in the sparsified graph $\frac{\log^2 n}{\epsilon^2} \sum_e \frac{1}{k'_e} = n \frac{\log^3 n}{\epsilon^2}$.

## Sampling using Nagamochi-Ibaraki Trees

- A collection of edge-disjoint forests.
- If $u$ and $v$ are connected in forest $i$, they are also connected in forests $1 \ldots i-1$.
- If edge $e = uv$ is in tree $i$, then $i = k'_e \leq k_e$.
- So sampling with probability $\frac{\log^2 n}{\epsilon^2 k'_e}$ preserves all cuts within $1 \pm \epsilon$ w.h.p.
- $\sum \frac{1}{k'_e} \leq n \log n$ (as opposed to $\sum \frac{1}{k_e} \leq n$)
- Expected number of edges in the sparsified graph $\frac{\log^2 n}{\epsilon^2} \sum_e \frac{1}{k'_e} = n\frac{\log^3 n}{\epsilon^2}$.

## Sampling using Nagamochi-Ibaraki Trees

- A collection of edge-disjoint forests.
- If $u$ and $v$ are connected in forest $i$, they are also connected in forests $1 \ldots i-1$.
- If edge $e = uv$ is in tree $i$, then $i = k'_e \leq k_e$.
- So sampling with probability $\frac{\log^2 n}{\epsilon^2 k'_e}$ preserves all cuts within $1 \pm \epsilon$ w.h.p.
- $\sum \frac{1}{k'_e} \leq n \log n$ (as opposed to $\sum \frac{1}{k_e} \leq n$)
- Expected number of edges in the sparsified graph $\frac{\log^2 n}{\epsilon^2} \sum_e \frac{1}{k'_e} = n \frac{\log^3 n}{\epsilon^2}$.

## Sampling using Nagamochi-Ibaraki Trees

- A collection of edge-disjoint forests.
- If $u$ and $v$ are connected in forest $i$, they are also connected in forests $1 \ldots i-1$.
- If edge $e = uv$ is in tree $i$, then $i = k'_e \leq k_e$.
- So sampling with probability $\frac{\log^2 n}{\epsilon^2 k'_e}$ preserves all cuts within $1 \pm \epsilon$ w.h.p.
- $\sum \frac{1}{k'_e} \leq n \log n$ (as opposed to $\sum \frac{1}{k_e} \leq n$)
- Expected number of edges in the sparsified graph $\frac{\log^2 n}{\epsilon^2} \sum_e \frac{1}{k'_e} = n \frac{\log^3 n}{\epsilon^2}$.

## Sampling using Nagamochi-Ibaraki Trees

- A collection of edge-disjoint forests.
- If $u$ and $v$ are connected in forest $i$, they are also connected in forests $1 \ldots i-1$.
- If edge $e = uv$ is in tree $i$, then $i = k'_e \leq k_e$.
- So sampling with probability $\frac{\log^2 n}{\epsilon^2 k'_e}$ preserves all cuts within $1 \pm \epsilon$ w.h.p.
- $\sum \frac{1}{k'_e} \leq n \log n$ (as opposed to $\sum \frac{1}{k_e} \leq n$)
- Expected number of edges in the sparsified graph $\frac{\log^2 n}{\epsilon^2} \sum_e \frac{1}{k'_e} = n\frac{\log^3 n}{\epsilon^2}$.

## Sampling using Nagamochi-Ibaraki Trees

- A collection of edge-disjoint forests.
- If $u$ and $v$ are connected in forest $i$, they are also connected in forests $1 \dots i-1$.
- If edge $e = uv$ is in tree $i$, then $i = k'_e \leq k_e$.
- So sampling with probability $\frac{\log^2 n}{\epsilon^2 k'_e}$ preserves all cuts within $1 \pm \epsilon$ w.h.p.
- $\sum \frac{1}{k'_e} \leq n \log n$ (as opposed to $\sum \frac{1}{k_e} \leq n$)
- Expected number of edges in the sparsified graph $\frac{\log^2 n}{\epsilon^2} \sum_e \frac{1}{k'_e} = n \frac{\log^3 n}{\epsilon^2}$.

## Sampling using Nagamochi-Ibaraki Trees

- A collection of edge-disjoint forests.
- If $u$ and $v$ are connected in forest $i$, they are also connected in forests $1 \ldots i-1$.
- If edge $e = uv$ is in tree $i$, then $i = k'_e \leq k_e$.
- So sampling with probability $\frac{\log^2 n}{\epsilon^2 k'_e}$ preserves all cuts within $1 \pm \epsilon$ w.h.p.
- $\sum \frac{1}{k'_e} \leq n \log n$ (as opposed to $\sum \frac{1}{k_e} \leq n$)
- Expected number of edges in the sparsified graph $\frac{\log^2 n}{\epsilon^2} \sum_e \frac{1}{k'_e} = n\frac{\log^3 n}{\epsilon^2}$.

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Define the $2^i$-projection of a cut to be the subset of its edges with $k'_e \sim 2^i$.
- Consider those cuts $\mathcal{C}$ where the size of the $2^i$-projection plus the size of $2^{i-1}$-projection is $\Delta_i$.
- We show that the number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- Note contrast from before where we had $n^{O(\frac{\Delta}{2^i})}$.

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Define the $2^i$-projection of a cut to be the subset of its edges with $k'_e \sim 2^i$.
- Consider those cuts $\mathcal{C}$ where the size of the $2^i$-projection plus the size of $2^{i-1}$-projection is $\Delta_i$.
- We show that the number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- Note contrast from before where we had $n^{O(\frac{\Delta}{2^i})}$.

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Define the $2^i$-projection of a cut to be the subset of its edges with $k'_e \sim 2^i$.
- Consider those cuts $\mathcal{C}$ where the size of the $2^i$-projection plus the size of $2^{i-1}$-projection is $\Delta_i$.
- We show that the number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- Note contrast from before where we had $n^{O(\frac{\Delta}{2^i})}$.

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Define the $2^i$-projection of a cut to be the subset of its edges with $k'_e \sim 2^i$.
- Consider those cuts $\mathcal{C}$ where the size of the $2^i$-projection plus the size of $2^{i-1}$-projection is $\Delta_i$.
- We show that the number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- Note contrast from before where we had $n^{O(\frac{\Delta}{2^i})}$.

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Define the $2^i$-projection of a cut to be the subset of its edges with $k'_e \sim 2^i$.
- Consider those cuts $\mathcal{C}$ where the size of the $2^i$-projection plus the size of $2^{i-1}$-projection is $\Delta_i$.
- We show that the number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- Note contrast from before where we had $n^{O(\frac{\Delta}{2^i})}$.

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Go back to sampling edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- The number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- For a particular $2^i$-projection $\mathcal{S}$,

$$Pr(||\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta_i) \leq n^{-\Theta(\frac{\Delta_i}{2^i})}$$

.

- For any given cut, $\sum_i \Delta_i \leq 2\Delta$.
- So every cut has deviation at most $2\epsilon\Delta$!

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Go back to sampling edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- The number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- For a particular $2^i$-projection $\mathcal{S}$,

$$Pr(||\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta_i) \leq n^{-\Theta(\frac{\Delta_i}{2^i})}$$

.
- For any given cut, $\sum_i \Delta_i \leq 2\Delta$.
- So every cut has deviation at most $2\epsilon\Delta$!

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Go back to sampling edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- The number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- For a particular $2^i$-projection $\mathcal{S}$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta_i) \leq n^{-\Theta(\frac{\Delta_i}{2^i})}$$

.

- For any given cut, $\sum_i \Delta_i \leq 2\Delta$.
- So every cut has deviation at most $2\epsilon\Delta$!

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Go back to sampling edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- The number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- For a particular $2^i$-projection $\mathcal{S}$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta_i) \leq n^{-\Theta(\frac{\Delta_i}{2^i})}$$

.

- For any given cut, $\sum_i \Delta_i \leq 2\Delta$.
- So every cut has deviation at most $2\epsilon\Delta$!

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Go back to sampling edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- The number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- For a particular $2^i$-projection $\mathcal{S}$,

$$Pr(||\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta_i) \leq n^{-\Theta(\frac{\Delta_i}{2^i})}$$

.
- For any given cut, $\sum_i \Delta_i \leq 2\Delta$.
- So every cut has deviation at most $2\epsilon\Delta$!

## Nagamochi-Ibaraki Sampling: Tighter Analysis

- Go back to sampling edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- The number of distinct $2^i$-projections over cuts in $\mathcal{C}$ is $n^{O(\frac{\Delta_i}{2^i})}$.
- For a particular $2^i$-projection $\mathcal{S}$,

$$Pr(|\mathcal{S}_{samp} - |\mathcal{S}|| \geq \epsilon\Delta_i) \leq n^{-\Theta(\frac{\Delta_i}{2^i})}$$

.

- For any given cut, $\sum_i \Delta_i \leq 2\Delta$.
- So every cut has deviation at most $2\epsilon\Delta$!

## Bounding the number of $2^i$-projections

- Take subgraph $G'$ formed by edges in NI trees $2^{i-2} \dots 2^i$.
- Key Property: An edge in NI trees $2^{i-1} \dots 2^i$ is at least $2^{i-2}$ connected in $G'$.
- So the number of $2^i$-projections in cuts of size $\Delta_i$ in $G'$ is $n^{O(\frac{\Delta_i}{2^{i-2}})}$, as needed.

## Bounding the number of $2^i$-projections

- Take subgraph $G'$ formed by edges in NI trees $2^{i-2} \ldots 2^i$.
- Key Property: An edge in NI trees $2^{i-1} \ldots 2^i$ is at least $2^{i-2}$ connected in $G'$.
- So the number of $2^i$-projections in cuts of size $\Delta_i$ in $G'$ is $n^{O(\frac{\Delta_i}{2^{i-2}})}$, as needed.

## Bounding the number of $2^i$-projections

- Take subgraph $G'$ formed by edges in NI trees $2^{i-2} \ldots 2^i$.
- Key Property: An edge in NI trees $2^{i-1} \ldots 2^i$ is at least $2^{i-2}$ connected in $G'$.
- So the number of $2^i$-projections in cuts of size $\Delta_i$ in $G'$ is $n^{O(\frac{\Delta_i}{2^{i-2}})}$, as needed.

## Bounding the number of $2^i$-projections

- Take subgraph $G'$ formed by edges in NI trees $2^{i-2} \ldots 2^i$.
- Key Property: An edge in NI trees $2^{i-1} \ldots 2^i$ is at least $2^{i-2}$ connected in $G'$.
- So the number of $2^i$-projections in cuts of size $\Delta_i$ in $G'$ is $n^{O(\frac{\Delta_i}{2^{i-2}})}$, as needed.

## Nagamochi-Ibaraki Tree Construction

- Process vertices in (a to be specified) order; for the chosen vertex, add all incident edges (these are incident on yet unprocessed vertices).
- For each vertex $v$, define $I(v)$ as the index of the first NI tree where $v$ is singleton.
- For each edge $e = uv$ processed, add $e$ to tree $min(I(u), I(v))$.
- Increment the smaller of $I(u), I(v)$ by 1; if both are equal, increment both.
- Successively pick the vertex with the largest $I()$ value for processing.

## Nagamochi-Ibaraki Tree Construction

- Process vertices in (a to be specified) order; for the chosen vertex, add all incident edges (these are incident on yet unprocessed vertices).
- For each vertex $v$, define $l(v)$ as the index of the first NI tree where $v$ is singleton.
- For each edge $e = uv$ processed, add $e$ to tree $min(l(u), l(v))$.
- Increment the smaller of $l(u), l(v)$ by 1; if both are equal, increment both.
- Successively pick the vertex with the largest $l()$ value for processing.

## Nagamochi-Ibaraki Tree Construction

- Process vertices in (a to be specified) order; for the chosen vertex, add all incident edges (these are incident on yet unprocessed vertices).
- For each vertex $v$, define $I(v)$ as the index of the first NI tree where $v$ is singleton.
- For each edge $e = uv$ processed, add $e$ to tree $min(I(u), I(v))$.
- Increment the smaller of $I(u), I(v)$ by 1; if both are equal, increment both.
- Successively pick the vertex with the largest $I()$ value for processing.

## Nagamochi-Ibaraki Tree Construction

- Process vertices in (a to be specified) order; for the chosen vertex, add all incident edges (these are incident on yet unprocessed vertices).
- For each vertex $v$, define $l(v)$ as the index of the first NI tree where $v$ is singleton.
- For each edge $e = uv$ processed, add $e$ to tree $min(l(u), l(v))$.
- Increment the smaller of $l(u), l(v)$ by 1; if both are equal, increment both.
- Successively pick the vertex with the largest $l()$ value for processing.

## Nagamochi-Ibaraki Tree Construction

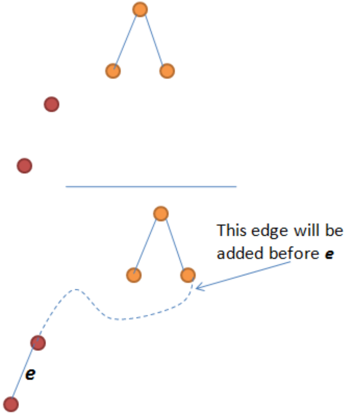
- Process vertices in (a to be specified) order; for the chosen vertex, add all incident edges (these are incident on yet unprocessed vertices).
- For each vertex $v$, define $l(v)$ as the index of the first NI tree where $v$ is singleton.
- For each edge $e = uv$ processed, add $e$ to tree $min(l(u), l(v))$.
- Increment the smaller of $l(u), l(v)$ by 1; if both are equal, increment both.
- Successively pick the vertex with the largest $l()$ value for processing.

## Nagamochi-Ibaraki Tree Construction

- Process vertices in (a to be specified) order; for the chosen vertex, add all incident edges (these are incident on yet unprocessed vertices).
- For each vertex $v$, define $I(v)$ as the index of the first NI tree where $v$ is singleton.
- For each edge $e = uv$ processed, add $e$ to tree $min(I(u), I(v))$.
- Increment the smaller of $I(u), I(v)$ by 1; if both are equal, increment both.
- Successively pick the vertex with the largest $I()$ value for processing.

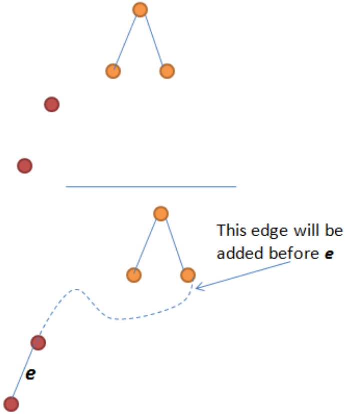
## Nagamochi-Ibaraki Tree Construction Contd.

- Key invariant: If a new connected component is created in a tree, it stays separate even after all future edge additions.
- $O(n \log n + m) \sim O(m)$ time.



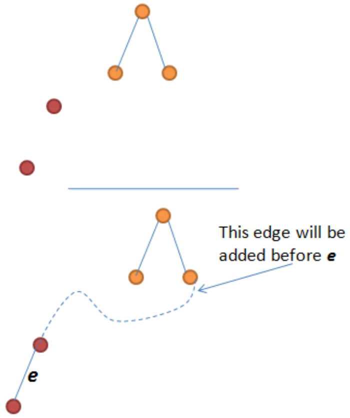
## Nagamochi-Ibaraki Tree Construction Contd.

- Key invariant: If a new connected component is created in a tree, it stays separate even after all future edge additions.
- $O(n \log n + m) \sim O(m)$ time.



## Nagamochi-Ibaraki Tree Construction Contd.

- Key invariant: If a new connected component is created in a tree, it stays separate even after all future edge additions.
- $O(n \log n + m) \sim O(m)$ time.



## Nagamochi-Ibaraki Sampling: Wrap Up

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- Every cut is preserved within a $1 \pm 2\epsilon$ factor, with inverse polynomial failure probability.

  The size of the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.
- The time taken for sampling is $O(n \log n + m)$.

## Nagamochi-Ibaraki Sampling: Wrap Up

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- Every cut is preserved within a $1 \pm 2\epsilon$ factor, with inverse polynomial failure probability.

  The size of the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.
- The time taken for sampling is $O(n \log n + m)$.

## Nagamochi-Ibaraki Sampling: Wrap Up

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- Every cut is preserved within a $1 \pm 2\epsilon$ factor, with inverse polynomial failure probability.

The size of the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.

- The time taken for sampling is $O(n \log n + m)$.

## Nagamochi-Ibaraki Sampling: Wrap Up

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- Every cut is preserved within a $1 \pm 2\epsilon$ factor, with inverse polynomial failure probability.

  The size of the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.
- The time taken for sampling is $O(n \log n + m)$.

## Nagamochi-Ibaraki Sampling: Wrap Up

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k'_e}$ ($k'_e$ is the index of the NI tree containing $e$).
- Every cut is preserved within a $1 \pm 2\epsilon$ factor, with inverse polynomial failure probability.

  The size of the sampled graph is $O(n\frac{\log^2 n}{\epsilon^2})$.
- The time taken for sampling is $O(n \log n + m)$.

## Effective Resistances

- The Effective Resistance $r_e$ of an edge $e$ is defined as follows:
- Treat the graph as a network of unit resistances.
- Push unit current into one endpoint of the edge, take unit current out of the other endpoint.
- What is the voltage drop across the edge? This is $r_e$.
- $r_e$ is also the fraction of spanning trees containing $e$.

## Effective Resistances

- The Effective Resistance $r_e$ of an edge $e$ is defined as follows:
- Treat the graph as a network of unit resistances.
- Push unit current into one endpoint of the edge, take unit current out of the other endpoint.
- What is the voltage drop across the edge? This is $r_e$.
- $r_e$ is also the fraction of spanning trees containing $e$.

## Effective Resistances

- The Effective Resistance $r_e$ of an edge $e$ is defined as follows:
- Treat the graph as a network of unit resistances.
- Push unit current into one endpoint of the edge, take unit current out of the other endpoint.
- What is the voltage drop across the edge? This is $r_e$.
- $r_e$ is also the fraction of spanning trees containing $e$.

## Effective Resistances

- The Effective Resistance $r_e$ of an edge $e$ is defined as follows:
- Treat the graph as a network of unit resistances.
- Push unit current into one endpoint of the edge, take unit current out of the other endpoint.
- What is the voltage drop across the edge? This is $r_e$.
- $r_e$ is also the fraction of spanning trees containing $e$.

## Effective Resistances

- The Effective Resistance $r_e$ of an edge $e$ is defined as follows:
- Treat the graph as a network of unit resistances.
- Push unit current into one endpoint of the edge, take unit current out of the other endpoint.
- What is the voltage drop across the edge? This is $r_e$.
- $r_e$ is also the fraction of spanning trees containing $e$.

## Effective Resistances

- The Effective Resistance $r_e$ of an edge $e$ is defined as follows:
- Treat the graph as a network of unit resistances.
- Push unit current into one endpoint of the edge, take unit current out of the other endpoint.
- What is the voltage drop across the edge? This is $r_e$.
- $r_e$ is also the fraction of spanning trees containing $e$.

## Sampling by Effective Conductance

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 c_e}$ (where $c_e = 1/r_e$).
- Key Property: $c_e \leq k_e$.
- Recall that underestimating $k_e$'s suffices.
- $\sum_e \frac{1}{c_e} = \sum_e r_e = n - 1$ (use the spanning tree fraction interpretation).
- So sampling with effective conductance yields a graph with $O(n\frac{\log^2 n}{\epsilon^2})$ edges that preserves all cuts within a $(1 \pm \epsilon)$ factor, w.h.p.

## Sampling by Effective Conductance

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 c_e}$ (where $c_e = 1/r_e$).
- Key Property: $c_e \leq k_e$.
- Recall that underestimating $k_e$'s suffices.
- $\sum_e \frac{1}{c_e} = \sum_e r_e = n - 1$ (use the spanning tree fraction interpretation).
- So sampling with effective conductance yields a graph with $O(n\frac{\log^2 n}{\epsilon^2})$ edges that preserves all cuts within a $(1 \pm \epsilon)$ factor, w.h.p.

## Sampling by Effective Conductance

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 c_e}$ (where $c_e = 1/r_e$).
- Key Property: $c_e \leq k_e$.
- Recall that underestimating $k_e$'s suffices.
- $\sum_e \frac{1}{c_e} = \sum_e r_e = n - 1$ (use the spanning tree fraction interpretation).
- So sampling with effective conductance yields a graph with $O(n\frac{\log^2 n}{\epsilon^2})$ edges that preserves all cuts within a $(1 \pm \epsilon)$ factor, w.h.p.

## Sampling by Effective Conductance

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 c_e}$ (where $c_e = 1/r_e$).
- Key Property: $c_e \leq k_e$.
- Recall that underestimating $k_e$'s suffices.
- $\sum_e \frac{1}{c_e} = \sum_e r_e = n - 1$ (use the spanning tree fraction interpretation).
- So sampling with effective conductance yields a graph with $O(n\frac{\log^2 n}{\epsilon^2})$ edges that preserves all cuts within a $(1 \pm \epsilon)$ factor, w.h.p.

## Sampling by Effective Conductance

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 c_e}$ (where $c_e = 1/r_e$).
- Key Property: $c_e \leq k_e$.
- Recall that underestimating $k_e$'s suffices.
- $\sum_e \frac{1}{c_e} = \sum_e r_e = n - 1$ (use the spanning tree fraction interpretation).
- So sampling with effective conductance yields a graph with $O(n\frac{\log^2 n}{\epsilon^2})$ edges that preserves all cuts within a $(1 \pm \epsilon)$ factor, w.h.p.

## Sampling by Effective Conductance

- Sample edge $e$ with probability $\frac{\log^2 n}{\epsilon^2 c_e}$ (where $c_e = 1/r_e$).
- Key Property: $c_e \leq k_e$.
- Recall that underestimating $k_e$'s suffices.
- $\sum_e \frac{1}{c_e} = \sum_e r_e = n - 1$ (use the spanning tree fraction interpretation).
- So sampling with effective conductance yields a graph with $O(n\frac{\log^2 n}{\epsilon^2})$ edges that preserves all cuts within a $(1 \pm \epsilon)$ factor, w.h.p.

## $c_e \leq k_e$, Why?

- Intuition: If the graph is just $k$ edge-disjoint paths between the endpoints of $e$, then each path has resistance at least 1, and $k_e$ paths pose a resistance of at least $1/k_e$. So $c_e \leq k_e$.
- But there are other edges around.
- Shrink these edges.
- Shrinking edge $f$ is like setting its resistance to 0, so effective resistance of $e$ should only decrease, i.e., conductance increases.
- Equivalently, given a random spanning tree $T$, $P(e \in T | f \in T) \leq P(e \in T)$. Rayleigh's monotonicity principle!

## $c_e \leq k_e$, Why?

- Intuition: If the graph is just $k$ edge-disjoint paths between the endpoints of $e$, then each path has resistance at least 1, and $k_e$ paths pose a resistance of at least $1/k_e$. So $c_e \leq k_e$.
- But there are other edges around.
- Shrink these edges.
- Shrinking edge $f$ is like setting its resistance to 0, so effective resistance of $e$ should only decrease, i.e., conductance increases.
- Equivalently, given a random spanning tree $T$, $P(e \in T | f \in T) \leq P(e \in T)$. Rayleigh's monotonicity principle!

## $c_e \leq k_e$, Why?

- Intuition: If the graph is just $k$ edge-disjoint paths between the endpoints of $e$, then each path has resistance at least 1, and $k_e$ paths pose a resistance of at least $1/k_e$. So $c_e \leq k_e$.
- But there are other edges around.
- Shrink these edges.
- Shrinking edge $f$ is like setting its resistance to 0, so effective resistance of $e$ should only decrease, i.e., conductance increases.
- Equivalently, given a random spanning tree $T$, $P(e \in T | f \in T) \leq P(e \in T)$. Rayleigh's monotonicity principle!

## $c_e \leq k_e$, Why?

- Intuition: If the graph is just $k$ edge-disjoint paths between the endpoints of $e$, then each path has resistance at least 1, and $k_e$ paths pose a resistance of at least $1/k_e$. So $c_e \leq k_e$.
- But there are other edges around.
- Shrink these edges.
- Shrinking edge $f$ is like setting its resistance to 0, so effective resistance of $e$ should only decrease, i.e., conductance increases.
- Equivalently, given a random spanning tree $T$, $P(e \in T | f \in T) \leq P(e \in T)$. Rayleigh's monotonicity principle!

## $c_e \leq k_e$, Why?

- Intuition: If the graph is just $k$ edge-disjoint paths between the endpoints of $e$, then each path has resistance at least 1, and $k_e$ paths pose a resistance of at least $1/k_e$. So $c_e \leq k_e$.
- But there are other edges around.
- Shrink these edges.
- Shrinking edge $f$ is like setting its resistance to 0, so effective resistance of $e$ should only decrease, i.e., conductance increases.
- Equivalently, given a random spanning tree $T$, $P(e \in T | f \in T) \leq P(e \in T)$. Rayleigh's monotonicity principle!

## $c_e \leq k_e$, Why?

- Intuition: If the graph is just $k$ edge-disjoint paths between the endpoints of $e$, then each path has resistance at least 1, and $k_e$ paths pose a resistance of at least $1/k_e$. So $c_e \leq k_e$.
- But there are other edges around.
- Shrink these edges.
- Shrinking edge $f$ is like setting its resistance to 0, so effective resistance of $e$ should only decrease, i.e., conductance increases.
- Equivalently, given a random spanning tree $T$, $P(e \in T | f \in T) \leq P(e \in T)$. Rayleigh's monotonicity principle!

## Proving Rayleigh's Monotonicity Principle: Energy

- A feasible flow is an assignment of current to the edges satisfying current conservation at each vertex, except the endpoints of $e$ which have a deficit/excess of 1, respectively.
- The energy of a feasible flow is $\sum_f i_f^2$ over all edges $f$.
- The energy of a feasible flow is also the voltage drop across $e$, which is the effective resistance of $e$ (easy proof using current conservation).
- Of all feasible flows, the one that minimizes energy has currents that are differences of endpoint voltages (can be shown using the primal-dual approach, for instance).

## Proving Rayleigh's Monotonicity Principle: Energy

- A feasible flow is an assignment of current to the edges satisfying current conservation at each vertex, except the endpoints of $e$ which have a deficit/excess of 1, respectively.
- The energy of a feasible flow is $\sum_f i_f^2$ over all edges $f$.
- The energy of a feasible flow is also the voltage drop across $e$, which is the effective resistance of $e$ (easy proof using current conservation).
- Of all feasible flows, the one that minimizes energy has currents that are differences of endpoint voltages (can be shown using the primal-dual approach, for instance).

## Proving Rayleigh's Monotonicity Principle: Energy

- A feasible flow is an assignment of current to the edges satisfying current conservation at each vertex, except the endpoints of $e$ which have a deficit/excess of 1, respectively.
- The energy of a feasible flow is $\sum_f i_f^2$ over all edges $f$.
- The energy of a feasible flow is also the voltage drop across $e$, which is the effective resistance of $e$ (easy proof using current conservation).
- Of all feasible flows, the one that minimizes energy has currents that are differences of endpoint voltages (can be shown using the primal-dual approach, for instance).

## Proving Rayleigh's Monotonicity Principle: Energy

- A feasible flow is an assignment of current to the edges satisfying current conservation at each vertex, except the endpoints of $e$ which have a deficit/excess of 1, respectively.
- The energy of a feasible flow is $\sum_f i_f^2$ over all edges $f$.
- The energy of a feasible flow is also the voltage drop across $e$, which is the effective resistance of $e$ (easy proof using current conservation).
- Of all feasible flows, the one that minimizes energy has currents that are differences of endpoint voltages (can be shown using the primal-dual approach, for instance).

## Proving Rayleigh's Monotonicity Principle: Energy

- A feasible flow is an assignment of current to the edges satisfying current conservation at each vertex, except the endpoints of $e$ which have a deficit/excess of 1, respectively.
- The energy of a feasible flow is $\sum_f i_f^2$ over all edges $f$.
- The energy of a feasible flow is also the voltage drop across $e$, which is the effective resistance of $e$ (easy proof using current conservation).
- Of all feasible flows, the one that minimizes energy has currents that are differences of endpoint voltages (can be shown using the primal-dual approach, for instance).

## Proof of Rayleigh's Monotonicity Principle

- If you shrink an edge $f$, then the least energy flow prior to shrinking $f$ is still a feasible flow after shrinking $f$.
- The least energy flow after shrinking $f$ then only reduces energy further.
- So the effective resistance of $e$ decreases when an edge $f$ is shrunk.

## Proof of Rayleigh's Monotonicity Principle

- If you shrink an edge $f$, then the least energy flow prior to shrinking $f$ is still a feasible flow after shrinking $f$.
- The least energy flow after shrinking $f$ then only reduces energy further.
- So the effective resistance of $e$ decreases when an edge $f$ is shrunk.

## Proof of Rayleigh's Monotonicity Principle

- If you shrink an edge $f$, then the least energy flow prior to shrinking $f$ is still a feasible flow after shrinking $f$.
- The least energy flow after shrinking $f$ then only reduces energy further.
- So the effective resistance of $e$ decreases when an edge $f$ is shrunk.

## Proof of Rayleigh's Monotonicity Principle

- If you shrink an edge $f$, then the least energy flow prior to shrinking $f$ is still a feasible flow after shrinking $f$.
- The least energy flow after shrinking $f$ then only reduces energy further.
- So the effective resistance of $e$ decreases when an edge $f$ is shrunk.

# Another Nagamochi-Ibaraki Sampling Scheme

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k''_e}$ ($k''_e$ is the index of the first NI tree where the endpoints of $e$ are not in the same connected component).
- Consider the graph $G''$ comprising edges $e$ with $k''_e \geq 2^{i-1}$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(k''_e)$ connected in $G''$.
- Replicate an edge in $G''$ with $k''_e \sim 2^j, j \geq i-1$, $n/2^j$ times, to obtain graph $H''$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(n)$ connected in $H''$.
- The number of distinct $2^i$-projections in cuts of size $X$ in $H''$ is $n^{O(\frac{X}{n})}$.

## Another Nagamochi-Ibaraki Sampling Scheme

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k''_e}$ ($k''_e$ is the index of the first NI tree where the endpoints of $e$ are not in the same connected component).
- Consider the graph $G''$ comprising edges $e$ with $k''_e \geq 2^{i-1}$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(k''_e)$ connected in $G''$.
- Replicate an edge in $G''$ with $k''_e \sim 2^j, j \geq i-1$, $n/2^j$ times, to obtain graph $H''$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(n)$ connected in $H''$.
- The number of distinct $2^i$-projections in cuts of size $X$ in $H''$ is $n^{O(\frac{X}{n})}$.

# Another Nagamochi-Ibaraki Sampling Scheme

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k''_e}$ ($k''_e$ is the index of the first NI tree where the endpoints of $e$ are not in the same connected component).
- Consider the graph $G''$ comprising edges $e$ with $k''_e \geq 2^{i-1}$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(k''_e)$ connected in $G''$.
- Replicate an edge in $G''$ with $k''_e \sim 2^j, j \geq i-1$, $n/2^j$ times, to obtain graph $H''$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(n)$ connected in $H''$.
- The number of distinct $2^i$-projections in cuts of size $X$ in $H''$ is $n^{O(\frac{X}{n})}$.

# Another Nagamochi-Ibaraki Sampling Scheme

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k''_e}$ ($k''_e$ is the index of the first NI tree where the endpoints of $e$ are not in the same connected component).
- Consider the graph $G''$ comprising edges $e$ with $k''_e \geq 2^{i-1}$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(k''_e)$ connected in $G''$.
- Replicate an edge in $G''$ with $k''_e \sim 2^j, j \geq i-1$, $n/2^j$ times, to obtain graph $H''$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(n)$ connected in $H''$.
- The number of distinct $2^i$-projections in cuts of size $X$ in $H''$ is $n^{O(\frac{X}{n})}$.

# Another Nagamochi-Ibaraki Sampling Scheme

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k''_e}$ ($k''_e$ is the index of the first NI tree where the endpoints of $e$ are not in the same connected component).
- Consider the graph $G''$ comprising edges $e$ with $k''_e \geq 2^{i-1}$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(k''_e)$ connected in $G''$.
- Replicate an edge in $G''$ with $k''_e \sim 2^j, j \geq i-1$, $n/2^j$ times, to obtain graph $H''$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(n)$ connected in $H''$.
- The number of distinct $2^i$-projections in cuts of size $X$ in $H''$ is $n^{O(\frac{X}{n})}$.

## Another Nagamochi-Ibaraki Sampling Scheme

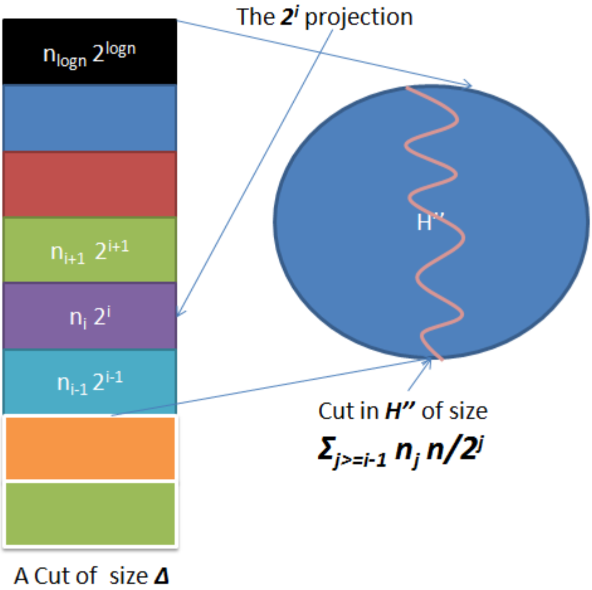
- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k''_e}$ ($k''_e$ is the index of the first NI tree where the endpoints of $e$ are not in the same connected component).
- Consider the graph $G''$ comprising edges $e$ with $k''_e \geq 2^{i-1}$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(k''_e)$ connected in $G''$.
- Replicate an edge in $G''$ with $k''_e \sim 2^j, j \geq i-1$, $n/2^j$ times, to obtain graph $H''$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(n)$ connected in $H''$.
- The number of distinct $2^i$-projections in cuts of size $X$ in $H''$ is $n^{O(\frac{X}{n})}$.

## Another Nagamochi-Ibaraki Sampling Scheme

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 k''_e}$ ($k''_e$ is the index of the first NI tree where the endpoints of $e$ are not in the same connected component).
- Consider the graph $G''$ comprising edges $e$ with $k''_e \geq 2^{i-1}$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(k''_e)$ connected in $G''$.
- Replicate an edge in $G''$ with $k''_e \sim 2^j, j \geq i - 1$, $n/2^j$ times, to obtain graph $H''$.
- Any edge $e$ with $k''_e \geq 2^i$ is $\Theta(n)$ connected in $H''$.
- The number of distinct $2^i$-projections in cuts of size $X$ in $H''$ is $n^{O(\frac{X}{n})}$.

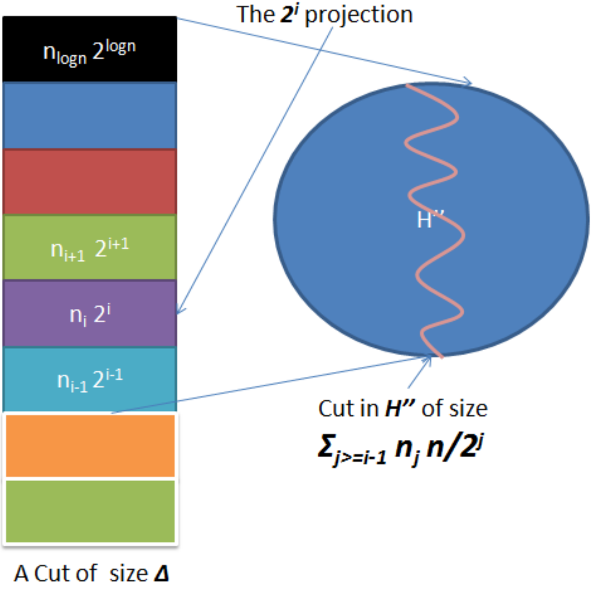
# Another Nagamochi-Ibaraki Sampling Scheme Contd.

- Consider one cut. How much deviation does the $2^i$-projection contribute?
- $\epsilon \sum_{j \geq i-1} \frac{n_j * \frac{n}{2^j}}{n} * 2^i = \epsilon \sum_{j \geq i-1} \frac{n_j}{2^{j-i}}$.
- Overall deviation $\epsilon \sum_{i>=0} \sum_{j \geq i-1} \frac{n_j}{2^{j-i}} = O(\epsilon \sum_{j \geq 0} n_j) = O(\epsilon \Delta)$



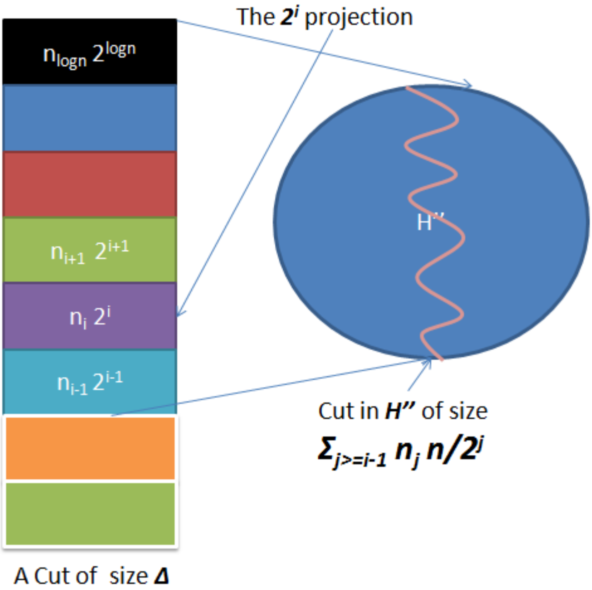
## Another Nagamochi-Ibaraki Sampling Scheme Contd.

- Consider one cut. How much deviation does the $2^i$-projection contribute?
- $\epsilon \sum_{j \geq i-1} \frac{n_j * \frac{n}{2^j}}{n} * 2^i = \epsilon \sum_{j \geq i-1} \frac{n_j}{2^{j-i}}$.
- Overall deviation $\epsilon \sum_{i>=0} \sum_{j \geq i-1} \frac{n_j}{2^{j-i}} = O(\epsilon \sum_{j \geq 0} n_j) = O(\epsilon \Delta)$



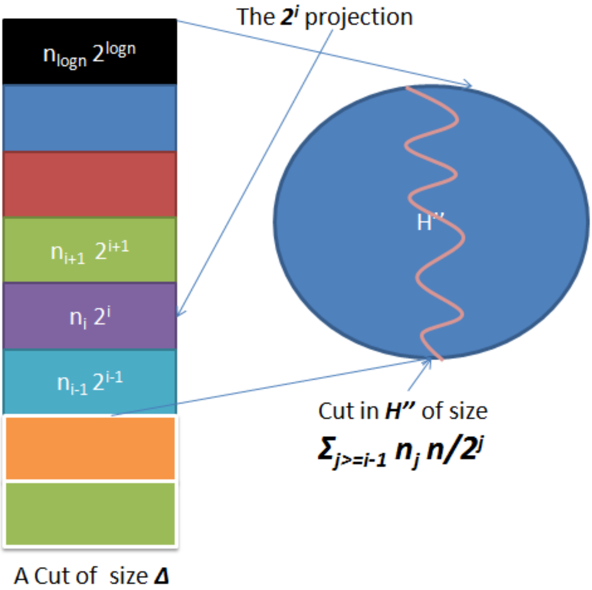
## Another Nagamochi-Ibaraki Sampling Scheme Contd.

- Consider one cut. How much deviation does the $2^i$-projection contribute?
- $\epsilon \sum_{j \geq i-1} \frac{n_j * \frac{n}{2^j}}{n} * 2^i = \epsilon \sum_{j \geq i-1} \frac{n_j}{2^{j-i}}$.
- Overall deviation $\epsilon \sum_{i>=0} \sum_{j \geq i-1} \frac{n_j}{2^{j-i}} = O(\epsilon \sum_{j \geq 0} n_j) = O(\epsilon \Delta)$



## Another Nagamochi-Ibaraki Sampling Scheme Contd.

- Consider one cut. How much deviation does the $2^i$-projection contribute?
- $\epsilon \sum_{j \geq i-1} \frac{n_j * \frac{n}{2^j}}{n} * 2^i = \epsilon \sum_{j \geq i-1} \frac{n_j}{2^{j-i}}$.
- Overall deviation $\epsilon \sum_{i>=0} \sum_{j \geq i-1} \frac{n_j}{2^{j-i}} = O(\epsilon \sum_{j \geq 0} n_j) = O(\epsilon \Delta)$



## Sampling by Strong Connectivity

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 sc_e}$ ($sc_e$ is the strong connectivity of $e$).
- Consider the graph $G'''$ comprising edges $e$ with $sc_e \geq 2^i$.
- Any edge $e$ with $sc_e \geq 2^i$ is $\Theta(sc_e)$ connected in $G'''$.
- So the same proof holds.
- $\sum_e sc_e \leq n-1$, so this yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier.

## Sampling by Strong Connectivity

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 sc_e}$ ($sc_e$ is the strong connectivity of $e$).
- Consider the graph $G'''$ comprising edges $e$ with $sc_e \geq 2^i$.
- Any edge $e$ with $sc_e \geq 2^i$ is $\Theta(sc_e)$ connected in $G'''$.
- So the same proof holds.
- $\sum_e sc_e \leq n-1$, so this yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier.

## Sampling by Strong Connectivity

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 sc_e}$ ($sc_e$ is the strong connectivity of $e$).
- Consider the graph $G'''$ comprising edges $e$ with $sc_e \geq 2^i$.
- Any edge $e$ with $sc_e \geq 2^i$ is $\Theta(sc_e)$ connected in $G'''$.
- So the same proof holds.
- $\sum_e sc_e \leq n - 1$, so this yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier.

## Sampling by Strong Connectivity

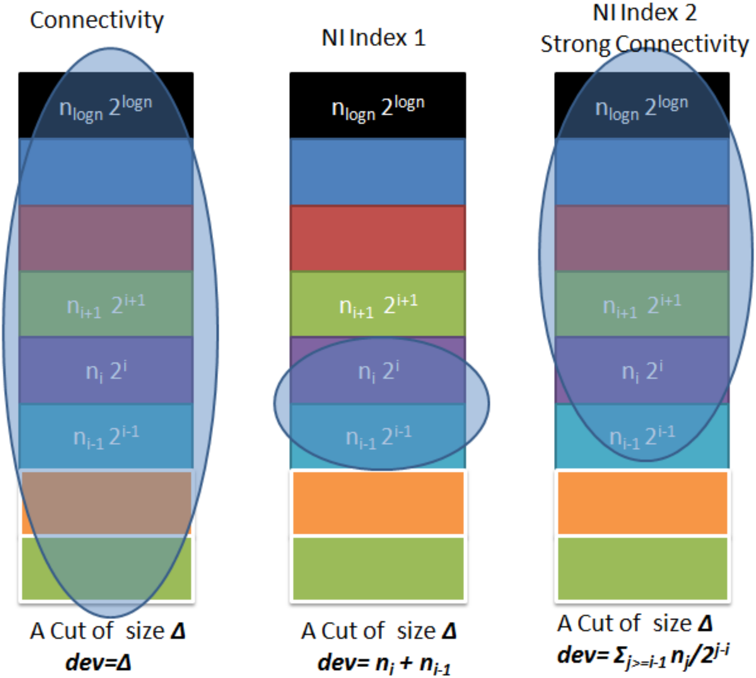
- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 sc_e}$ ($sc_e$ is the strong connectivity of $e$).
- Consider the graph $G'''$ comprising edges $e$ with $sc_e \geq 2^i$.
- Any edge $e$ with $sc_e \geq 2^i$ is $\Theta(sc_e)$ connected in $G'''$.
- So the same proof holds.
- $\sum_e sc_e \leq n - 1$, so this yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier.

## Sampling by Strong Connectivity

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 sc_e}$ ($sc_e$ is the strong connectivity of $e$).
- Consider the graph $G'''$ comprising edges $e$ with $sc_e \geq 2^i$.
- Any edge $e$ with $sc_e \geq 2^i$ is $\Theta(sc_e)$ connected in $G'''$.
- So the same proof holds.
- $\sum_e sc_e \leq n-1$, so this yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier.

## Sampling by Strong Connectivity

- Sample edge $e$ with probability $\frac{\log n}{\epsilon^2 sc_e}$ ($sc_e$ is the strong connectivity of $e$).
- Consider the graph $G'''$ comprising edges $e$ with $sc_e \geq 2^i$.
- Any edge $e$ with $sc_e \geq 2^i$ is $\Theta(sc_e)$ connected in $G'''$.
- So the same proof holds.
- $\sum_e sc_e \leq n - 1$, so this yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier.


Connectivity
NI Index 1
NI Index 2
Strong Connectivity
$n_{logn}\ 2^{logn}$
$n_{i+1}\ 2^{i+1}$
$n_i\ 2^i$
$n_{i-1}\ 2^{i-1}$
A Cut of size $\Delta$
$dev=\Delta$
A Cut of size $\Delta$
$dev= n_i + n_{i-1}$
A Cut of size $\Delta$
$dev= \Sigma_{j>=i-1}\ n_j/2^{j-i}$

## Other Results

- An $O(n\frac{\log n}{\epsilon^2})$ size sparsifier in time $O(n\log n + m)$ (Hariharan and Panigrahy)
- Sampling by conductance yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier (Spielman, Srivastava); this is more general as well, but conductances are more complex to compute.
- An $O(\frac{n}{\epsilon^2})$ size sparsifier (Batson, Spielman, Srivastava).

## Other Results

- An $O(n\frac{\log n}{\epsilon^2})$ size sparsifier in time $O(n \log n + m)$ (Hariharan and Panigrahy)
- Sampling by conductance yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier (Spielman, Srivastava); this is more general as well, but conductances are more complex to compute.
- An $O(\frac{n}{\epsilon^2})$ size sparsifier (Batson, Spielman, Srivastava).

## Other Results

- An $O(n\frac{\log n}{\epsilon^2})$ size sparsifier in time $O(n \log n + m)$ (Hariharan and Panigrahy)
- Sampling by conductance yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier (Spielman, Srivastava); this is more general as well, but conductances are more complex to compute.
- An $O(\frac{n}{\epsilon^2})$ size sparsifier (Batson, Spielman, Srivastava).

## Other Results

- An $O(n\frac{\log n}{\epsilon^2})$ size sparsifier in time $O(n \log n + m)$ (Hariharan and Panigrahy)
- Sampling by conductance yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier (Spielman, Srivastava); this is more general as well, but conductances are more complex to compute.
- An $O(\frac{n}{\epsilon^2})$ size sparsifier (Batson, Spielman, Srivastava).

## Open Problem

- Show that sampling by connectivity yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier, w.h.p.
- This will yield a corresponding corollary for sampling by conductances.

## Open Problem

- Show that sampling by connectivity yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier, w.h.p.
- This will yield a corresponding corollary for sampling by conductances.

## Open Problem

- Show that sampling by connectivity yields an $O(n\frac{\log n}{\epsilon^2})$ size sparsifier, w.h.p.
- This will yield a corresponding corollary for sampling by conductances.

- THANK YOU

- THAT YOU