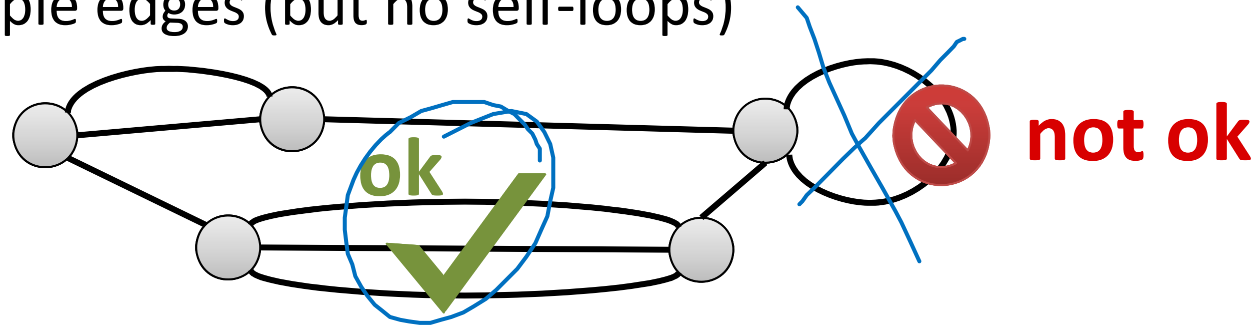
Chapter 6
Randomization

Algorithm Theory
WS 2012/13

Fabian Kuhn

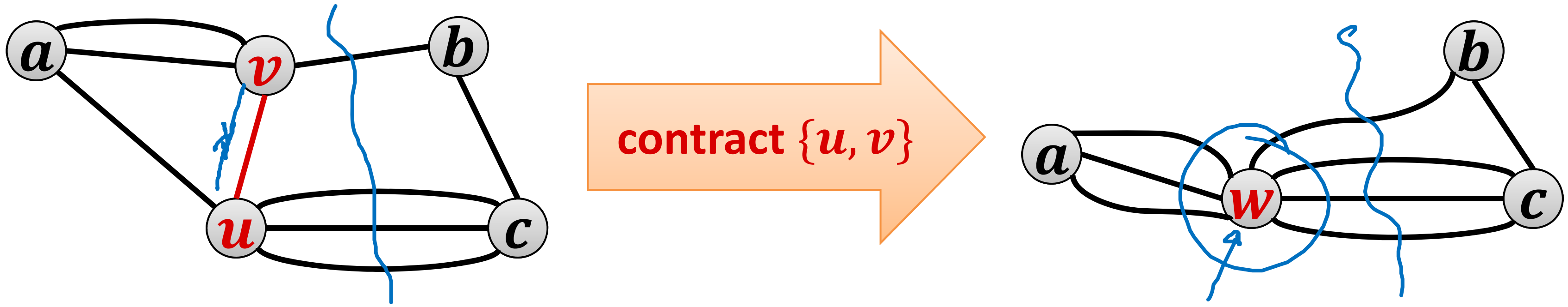
Edge Contractions

- In the following, we consider multi-graphs that can have multiple edges (but no self-loops)



Contracting edge $\{u, v\}$:

- Replace nodes u, v by new node w
- For all edges $\{u, x\}$ and $\{v, x\}$, add an edge $\{w, x\}$
- Remove self-loops created at node w



Randomized Contraction Algorithm

Algorithm:

while there are >2 nodes **do**

contract a uniformly random edge

return cut induced by the last two remaining nodes

(cut defined by the original node sets represented by the last 2 nodes)

$A \quad B$

(A, B)

Theorem: The random contraction algorithm returns a minimum cut with probability at least $2/n(n-1)$.

$1/\binom{n}{2}$

- We showed this last week.

$2/O(n^2)$

Theorem: The random contraction algorithm can be implemented in time $O(n^2)$.

- There are $n-2$ contractions, each can be done in time $O(n)$.
- You will show this in the exercises.

Randomized Min Cut Algorithm

Theorem: If the contraction algorithm is repeated $O(n^2 \log n)$ times, one of the $O(n^2 \log n)$ instances returns a min. cut w.h.p.

Proof:

$O(n^2 \log n)$

- Probability to not get a minimum cut in $c \cdot \binom{n}{2} \cdot \ln n$ iterations:

$1 - x \leq e^x$

$< e^{-1/\binom{n}{2}}$

$$\underbrace{\left(1 - \frac{1}{\binom{n}{2}}\right)}_{\text{no min. cut in one instance}}^{c \cdot \binom{n}{2} \cdot \ln n} < e^{-c \ln n} = \frac{1}{n^c}$$

Corollary: The contraction algorithm allows to compute a minimum cut in $O(n^4 \log n)$ time w.h.p.

- Each instance can be implemented in $O(n^2)$ time.
  ($O(n)$ time per contraction)

Can We Do Better?

- Time $O(n^4 \log n)$ is not very spectacular, a simple max flow based implementation has time $O(n^4)$.

However, we will see that the contraction algorithm is nevertheless very interesting because:

1. The algorithm can be improved to beat every known deterministic algorithm.

1. It allows to obtain strong statements about the distribution of cuts in graphs.

Better Randomized Algorithm

Recall:

- Consider a fixed min cut (A, B), assume (A, B) has size k
- The algorithm outputs (A, B) iff none of the k edges crossing (A, B) gets contracted.
- Throughout the algorithm, the edge connectivity is at least k and therefore each node has degree $\geq k$ $\longrightarrow$ #edges $\geq \frac{k \cdot \#\text{nodes}}{2}$
- Before contraction i, there are $n + 1 - i$ nodes and thus at least $(n + 1 - i)k/2$ edges
- If no edge crossing (A, B) is contracted before, the probability to contract an edge crossing (A, B) in step i is at most

$$\frac{\cancel{k}}{\frac{(n+1-i)\cancel{k}}{2}} = \frac{2}{n+1-i}.$$

Improving the Contraction Algorithm

- For a specific min cut (A, B), if (A, B) survives the first i contractions,

$$\mathbb{P}(\text{edge crossing } (A,B) \text{ in contraction } i+1) \leq \frac{2}{n-i}.$$

- **Observation:** The probability only gets large for large i

- **Idea:** The early steps are much safer than the late steps.

  Maybe we can repeat the late steps more often than the early ones.

maybe we can get a better success prob. with less work

Safe Contraction Phase

Lemma: A given min cut (A, B) of an n-node graph G survives the first $\underbrace{n - \left\lceil n/\sqrt{2} + 1 \right\rceil}_{t}$ contractions, with probability $> 1/2$.

Proof:

$$\mathbb{P}(\mathcal{E}_{i+1} \mid \mathcal{E}_1 \cap \dots \cap \mathcal{E}_i) \geq 1 - \frac{2}{n-i} = \frac{n-i-2}{n}$$

- Event $\mathcal{E}_i$: cut (A, B) survives contraction i
- Probability that (A, B) survives the first $n - t$ contractions:

$$\underbrace{\left(1-\frac{2}{n}\right)}_{\mathbb{P}(\mathcal{E}_1)}\underbrace{\left(1-\frac{2}{n-1}\right)}_{\mathbb{P}(\mathcal{E}_2 \mid \mathcal{E}_1)}\cdot \ldots \cdot \underbrace{\left(1-\frac{2}{t+1}\right)}_{\mathbb{P}(\mathcal{E}_{n-t} \mid \mathcal{E}_1 \cap \dots \cap \mathcal{E}_{n-t-1})} = \frac{n-2}{n}\cdot\frac{n-3}{n-1}\cdot\frac{n-4}{n-2}\cdot \ldots \cdot\frac{t-1}{t+1}$$

$$\frac{\cancel{n-2}}{n}\cdot\frac{\cancel{n-3}}{n-1}\cdot\frac{\cancel{n-4}}{\cancel{n-2}}\cdot \ldots \cdot\frac{t}{\cancel{t+2}}\cdot\frac{t-1}{\cancel{t+1}} = \frac{t(t-1)}{n(n-1)}$$

$$\frac{\left\lceil n/\sqrt{2} + 1\right\rceil \left\lceil n/\sqrt{2}\right\rceil}{n(n-1)} \geq \frac{\left(\frac{n}{\sqrt{2}}+1\right)\left(\frac{n}{\sqrt{2}}\right)}{n(n-1)} = \underbrace{\frac{\frac{n}{\sqrt{2}}+1}{n}}_{> 1/\sqrt{2}} \cdot \underbrace{\frac{\frac{n}{\sqrt{2}}}{n-1}}_{> 1/\sqrt{2}} > \frac{1}{2}$$

Better Randomized Algorithm

Let's simplify a bit:

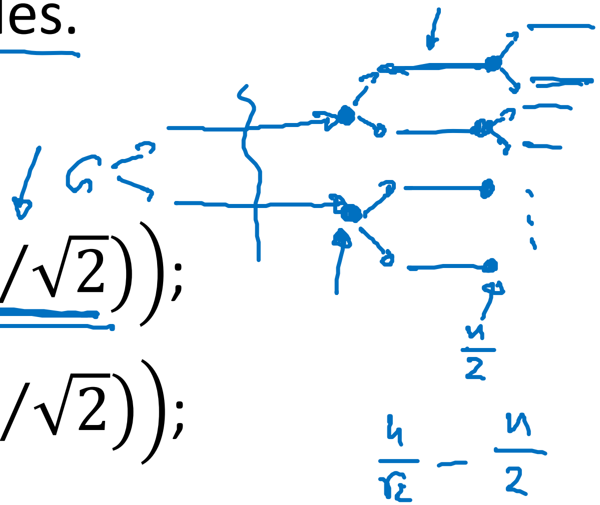
- Pretend that $n/\sqrt{2}$ is an integer (for all n we will need it).
- Assume that a given min cut survives the first $n - {}^{n}/_{\sqrt{2}}$ contractions with probability $\geq {}^{1}/_{2}$.

$\mathbf{contract}(\boldsymbol{G}, \boldsymbol{t})$**:** $t = \frac{n}{\sqrt{2}}$

- Starting with n-node graph G, perform $n - t$ edge contractions such that the new graph has t nodes.

$\mathbf{mincut}(\boldsymbol{G})$**:**

1. $X_1 \coloneqq \mathrm{mincut}\Big(\mathrm{contract}\big(G, n/\sqrt{2}\big)\Big);$
2. $X_2 \coloneqq \mathrm{mincut}\Big(\mathrm{contract}\big(G, n/\sqrt{2}\big)\Big);$
3. **return** $\min\{X_1, X_2\};$



Success Probability

mincut(G):

contains min cut with prob $\geq 1/2$

1. $X_1 := \text{mincut}\left(\text{contract}(G, n/\sqrt{2})\right)$;
2. $X_2 := \text{mincut}\left(\text{contract}(G, n/\sqrt{2})\right)$;
3. **return** $\min\{X_1, X_2\}$;

$1-(1-q)^2$

$\boldsymbol{P(n)}$: probability that the above algorithm returns a min cut when applied to a graph with n nodes.

- Probability that X_1 is a min cut $\geq \underbrace{\frac{1}{2}\cdot P(n/\sqrt{2})}_{q}$

Recursion:

$$P(n) \geq 1-\left(1-\frac{1}{2}P(n/\sqrt{2})\right)^2 = 1-\left(1-P(n/\sqrt{2})+\frac{1}{4}\cdot P(n/\sqrt{2})^2\right) = P(n/\sqrt{2})-\frac{1}{4}P(n/\sqrt{2})^2$$

Success Probability

$\log_{\sqrt{2}}(1/\sqrt{2}) = -\frac{1}{2}$

Theorem: The recursive randomized min cut algorithm returns a minimum cut with probability at least $1/\log_2 n$.

Proof (by induction on n):

$$P(n) \geq P\left(\frac{n}{\sqrt{2}}\right) - \frac{1}{4} \cdot P\left(\frac{n}{\sqrt{2}}\right)^2, \qquad P(2) = 1$$

Base case $P(2) = 1 \geq \frac{1}{\log 2} = 1$

$$P(n) \geq \frac{1}{\log \frac{n}{\sqrt{2}}} - \frac{1}{4\left(\log\left(\frac{n}{\sqrt{2}}\right)\right)^2}$$

$$= \frac{1}{\log n - \frac{1}{2}} - \frac{1}{4\left(\log n - \frac{1}{2}\right)^2}$$

$$= \frac{4\left(\log n - \frac{1}{2}\right) - 1}{4\left(\log n - \frac{1}{2}\right)^2} = \frac{4\log n - 3}{4\log^2 n - 4\log n + 1} = \frac{4\log n - 4 + \frac{1}{\log n} + 1 - \frac{1}{\log n}}{4\log^2 n - 4\log n + 1}$$

$$= \frac{1}{\log n} + \underbrace{\frac{1 - \frac{1}{\log n}}{\cdots}}_{>0} > \frac{1}{\log n}$$

Running Time

1. $X_1 := \text{mincut}\left(\text{contract}\left(G, n/\sqrt{2}\right)\right);$
2. $X_2 := \text{mincut}\left(\text{contract}\left(G, n/\sqrt{2}\right)\right);$
3. **return** $\min\{X_1, X_2\};$

Recursion:

- $T(n)$: time to apply algorithm to n-node graphs
- Recursive calls: $2T\left({}^{n}/_{\sqrt{2}}\right)$
- Number of contractions to get to ${}^{n}/_{\sqrt{2}}$ nodes: $O(n)$

time $O(n^2)$

$$T(n) = 2T\left(\frac{n}{\sqrt{2}}\right) + O(n^2), \qquad T(2) = O(1)$$

Running Time

Theorem: The running time of the recursive, randomized min cut algorithm is $O(n^2 \log n)$.

Proof:

- Can be shown in the usual way, by induction on n

Remark:

- The running time is only by an $O(\log n)$-factor slower than the basic contraction algorithm.
- The success probability is exponentially better!

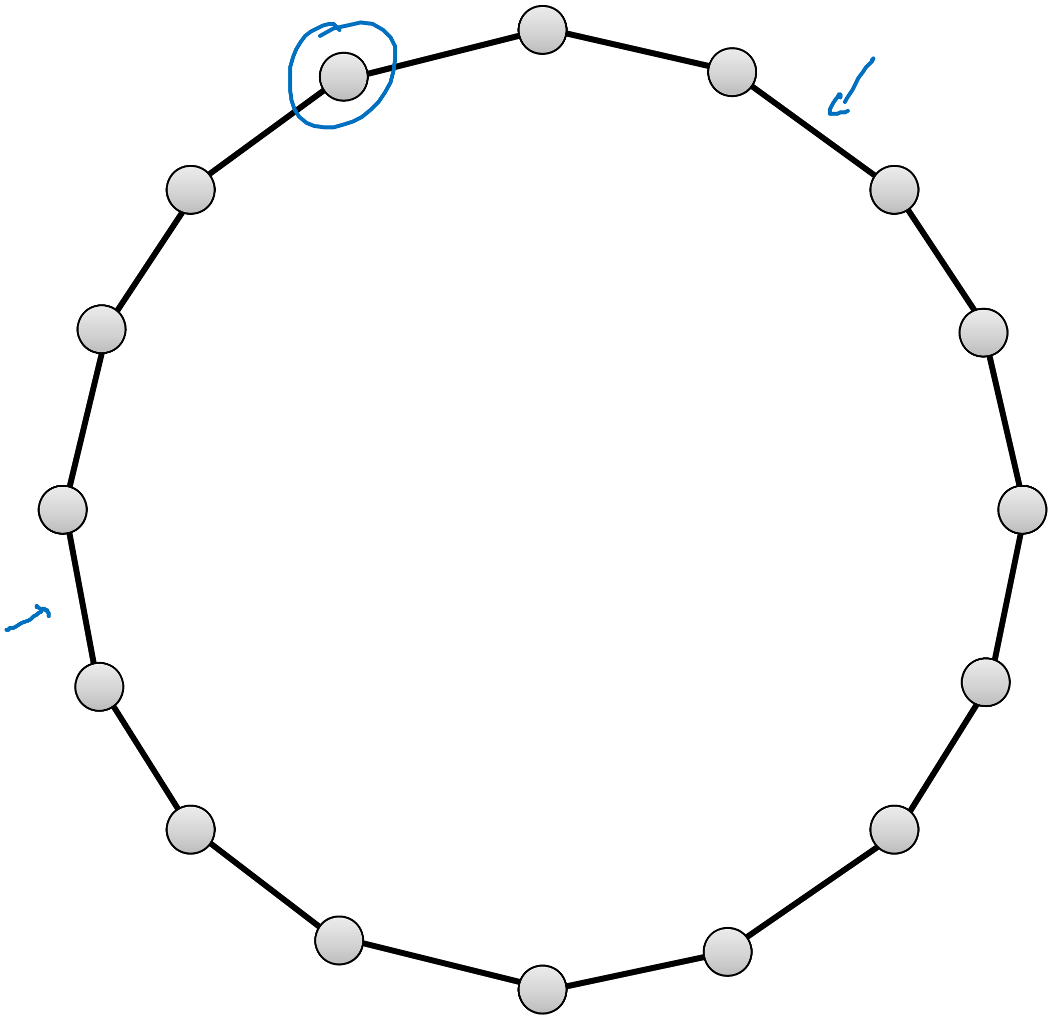
Number of Minimum Cuts

- Given a graph G, how many minimum cuts can there be?
- Or alternatively: If G has edge connectivity k, how many ways are there to remove k edges to disconnect G?
- Note that the total number of cuts is large.

$\# \text{cuts} \approx 2^{n-1}$

Number of Minimum Cuts

Example: Ring with n nodes



- Minimum cut size: 2
- Every two edges induce a min cut
- Number of edge pairs:

$$\binom{n}{2}$$

- Are there graphs with more min cuts?

Number of Min Cuts

Theorem: The number of minimum cuts of a graph is at most $\binom{n}{2}$.

Proof:

- Assume there are s min cuts
- For $i \in \{1, \dots, s\}$, define event $\mathcal{C}_i$:

  $\mathcal{C}_i :=$ {basic contraction algorithm returns min cut i}
- We know that for $i \in \{1, \dots, s\}$: $\mathbb{P}(\mathcal{C}_i) \geq 1/\binom{n}{2}$
- Events $\mathcal{C}_1, \dots, \mathcal{C}_s$ are disjoint:

$$1 \geq \mathbb{P}\left(\bigcup_{i=1}^{s} \mathcal{C}_i\right) = \sum_{i=1}^{s} \mathbb{P}(\mathcal{C}_i) \geq \frac{s}{\binom{n}{2}} \implies s \leq \binom{n}{2}$$

Counting Larger Cuts

- In the following, assume that min cut has size $\lambda = \lambda(G)$
- How many cuts of size $k \leq \alpha \cdot \lambda$ can a graph have?
- Consider a specific cut (A, B) of size $\leq k$
- As before, during the contraction algorithm:
  - min cut size $\geq \lambda$
  - number of edges $\geq \boxed{\lambda \cdot \#\text{nodes}/2}$
  - cut (A, B) remains (and has size k) as long as no edge of it gets contracted
- Prob. that an edge crossing (A, B) is chosen in i^{th} contraction

$$= \frac{k}{\#\text{edges}} \leq \frac{2k}{\lambda \cdot \#\text{nodes}} = \frac{2\alpha}{n - i + 1}$$

Counting Larger Cuts

Lemma: The probability that cut (A, B) of size $\alpha \cdot \lambda$ survives the first $n - 2\alpha$ edge contractions is at least

$$\frac{(2\alpha)!}{n(n-1)\cdot \ldots \cdot (n-2\alpha+1)} \geq \frac{2^{2\alpha-1}}{n^{2\alpha}}.$$

Proof:

- As before, event $\mathcal{E}_i$: cut (A, B) survives contraction i

$\mathbb{P}((A,B) \text{ survives } n-2\alpha \text{ contr.}) \geq$

$$\frac{n-2\alpha}{n} \cdot \frac{n-2\alpha-1}{n-1} \cdot \frac{n-2\alpha-2}{n-2} \cdot \ldots \cdot \frac{1}{2\alpha+1}$$

$$= \frac{2\alpha \cdot (2\alpha-1) \cdot \ldots \cdot 1}{n(n-1)\cdot \ldots \cdot (n-2\alpha+1)} = \frac{(2\alpha)!}{(n-1)\cdot \ldots \cdot (n-2\alpha+1)} \geq \frac{2^{2\alpha-1}}{n^{2\alpha}}$$

Number of Cuts

Theorem: The number of edge cuts of size at most $\alpha \cdot \lambda(G)$ in an n-node graph G is at most $\cancel{2}n^{2\alpha}$.

Proof:

- specific cut (A,B) of size $\alpha \cdot \lambda$ survives first $n-2\alpha$ contr. with prob. $\geq \frac{2^{2\alpha-1}}{n^{2\alpha}}$ $\Longrightarrow$ graph with 2α nodes
- return a uniform random remaining cut
- # remaining cuts $\leq 2^{2\alpha-1}$
- Prob. to return $(A,B) \geq \frac{1}{n^{2\alpha}}$

# of such cuts $\leq n^{2\alpha}$