# $\left\{ \text{Csc 80030} \mid \text{Lecture 6} \right\}$

PROBABILISTIC ANALYSIS &

RANDOMIZED ALGORITHMS

Hoeteck Wee · hoeteck@cs.qc.edu

## PART 0

- HOMEWORK: HW3 due next week
- PROGRAMMING ASSIGNMENT: is out, due Mar 25
- TODAY: probabilistic method & derandomization

# PART 1 | derandomization

# Derandomization

- NAMELY, remove the randomness from randomized algorithms.
  - Eliminate errors, from *w.h.p.* to *always*
  - Theoretical interest: does randomness *really* help in the design of algorithms?
- TWO TECHNIQUES. method of conditional expectations & pairwise independence.

# Finding a Large Cut

- THEOREM: Any graph $G$ with $m$ edges has a cut of size $\geq \frac{m}{2}$.

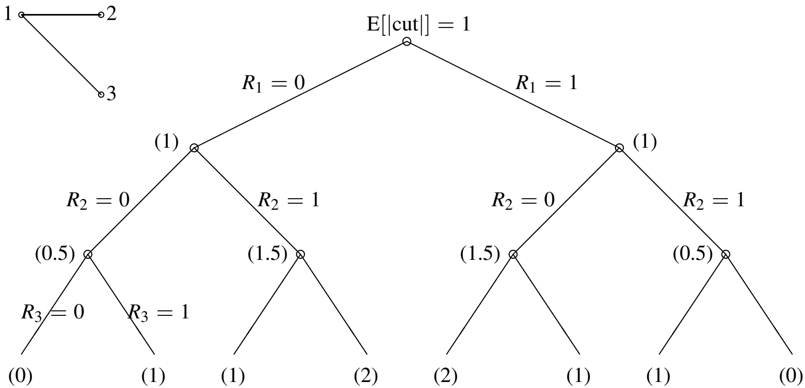
**LARGECUT Algorithm**

Input: a graph $G = ([n], E)$

1. Flip $n$ coins $r_1, r_2, \ldots, r_n$, put vertex $i$ in $S$ if $r_i = 0$ and in $T$ otherwise.
2. Output $(S, T)$.

- by averaging argument, there exists a choice of $r_1, r_2, \ldots, r_n$ that leads to a cut of size at least $\frac{m}{2}$.
- find a good sequence of coin tosses “bit by bit”.

# LARGECUT tree




QUESTION. How to compute labels of the “internal” nodes?

# Deterministic LargeCut

- DEFINITION. Define the conditional expectation

$$e(r_1, r_2, \ldots, r_i) = \mathrm{E}_{R_1,\ldots,R_n}\Big[|\mathrm{cut}(S,T)| \mid R_1 = r_1, R_2 = r_2, \ldots, R_i = r_i\Big]$$

- BASE CASE. $e(\lambda) = |E|/2$.
- INDUCTIVE CASE. $e(\lambda) = \mathrm{E}_{R_1}[e(R_1)]$.

  More generally, $e(r_1, \ldots, r_i) = \mathrm{E}_{R_{i+1}}[e(r_1, \ldots, r_i, R_{i+1})]$
- CLAIM. There exists $r_1, r_2, \ldots, r_n$ such that

  $e(\lambda) \leq e(r_1) \leq e(r_1, r_2) \leq \cdots \leq e(r_1, r_2, \ldots, r_n)$
- Let $S_i = \{j : j \leq i, r_j = 0\}$ and $T_i = \{j : j \leq i, r_j = 1\}$. Then,

$$e(r_1, r_2, \ldots, r_i) = |\mathrm{cut}(S_i, T_i)| + \tfrac{1}{2}|\mathrm{cut}(\{i+1, \ldots, n\}, [n])|$$

# Deterministic LARGECUT

## Deterministic LARGECUT Algorithm I

1. Set $S = \emptyset, T = \emptyset$
2. For $i = 0, \ldots, n-1$:
   2.1 If $|\mathrm{cut}(\{i+1\}, S)| > |\mathrm{cut}(\{i+1\}, \bar{S})|$, set $T \leftarrow T \cup \{i+1\}$,
   2.2 else set $S \leftarrow S \cup \{i+1\}$.

- REMARK. This is the "natural" greedy algorithm. Method of conditional expectations tells us which objective function to optimize locally.

# Derandomization via pairwise independence

- ANALYSIS, REVISITED.

$$\mathrm{E}[|\mathrm{cut}(S,T)|] = \sum_{(i,j)\in E} \Pr[R_i \neq R_j] = |E|/2$$

- OBSERVATION. Applies for any distribution on r.v.'s $(R_1, \ldots, R_n)$ satisfying $\Pr[R_i \neq R_j] = 1/2$ for each $i \neq j$; that is, *pairwise independent*
  - each $R_i$ is an unbiased random bit;
  - for each $i \neq j$, $R_i$ is independent from $R_j$
- QUESTION. Can we generate $N$ pairwise independent bits using less than $N$ truly random bits?

# Pairwise independent bits

- CONSTRUCTION. Let $B_1, \ldots, B_k$ be $k$ independent unbiased random bits. For each nonempty set $A \subseteq [k]$, let $R_A$ be the r.v. $\oplus_{i \in A} B_i$.
- CLAIM. The $2^k - 1$ random variables $R_A$ are pairwise independent unbiased random bits.
  - Clear that each $R_A$ is unbiased.
  - For pairwise independence, consider any $A \neq A' \subseteq [k]$. A'hen,

$$R_A = R_{A \cap A'} \oplus R_{A \setminus A'}; \quad R_{A'} = R_{A \cap A'} \oplus R_{A' \setminus A}$$

  - $R_{A \cap A'}, R_{A \setminus A'}, R_{A' \setminus A}$ are independent and at least two are non-empty.
  - Hence, $R_A, R_{A'}$ takes each value in $\{0, 1\}^2$ with prob. $1/4$.
- Can generate $N$ pairwise independent bits from $\lceil \log(N+1) \rceil$ independent random bits.

# Deterministic LargeCut, II

## Deterministic LargeCut Algorithm II

1. For all sequences of bits $b_1, b_2, \ldots, b_{\lceil \log(n+1) \rceil}$, run the randomized LargeCut algorithm using coin tosses $(r_A = \oplus_{i \in A} b_i)_{A \neq \emptyset}$ and choose the largest cut thus obtained.

# PART 2 | sample and modify

# Sample and Modify

- STAGE ONE. Construct a random structure that does not have the required properties.
- STAGE TWO. Modify the structure to have the required property.
- APPLICATION. Obtain bounds on the size of the largest independent set in a graph (set of vertices with no edges between them).

# Existence of large independent sets

- THEOREM: Any graph $G$ with $n$ vertices and $m$ edges has an independent set with at least $n^2/4m$ vertices.
- CONSTRUCTION.
  1. Delete each vertex of $G$ (and its incident edges) with prob $1 - p$.
  2. For each remaining edge, remove it and one of its adjacent vertices.
- CLAIM. Let $X$ and $Y$ be resp. the # of vertices and edges that survive step 1. Then, construction outputs an I.S. of size at least $X - Y$.
- ANALYSIS. $\mathrm{E}[X] = np$ and $\mathrm{E}[Y] = mp^2$, so $\mathrm{E}[X - Y] = np - mp^2$. maximized at $n - 2mp = 0$, i.e. $p = 2m/n$ and $\mathrm{E}[X - Y] = n^2/4m$.

THE END | next, power of two choices