CS 6815: Pseudorandomness and Combinatorial Constructions **Fall 2022**

Lecture 4: Sep 01, 2022

Lecturer: Eshan Chattopadhyay *Scribe: Jenny Chen*

4.1 Max-Cut Problem

The Max-Cut Problem is described as the following: given an undirected, unweighted graph $G = (V, E)$, want to find a cut $S \subseteq V$ which maximizes $\text{cut}(S) = \{(u, v) \in E |$ exactly one of u, v is in S$\}$.

The corresponding decision problem is: Given a undirected and unweighted graph $G = (V, E)$ and an integer k, we want to know whether there exist a cut S which $\text{cut}(S) \geq k$. This is well-known to be NP-complete.

We give a simple randomized 1/2-approximation algorithm for the Max-Cut Problem.

Algorithm 4.1. *Given input $G = (V, E)$, for each $v \in V$, we place v in S with a probability of $1/2$ (each decision is made independently).*

Clearly the algorithm runs in $O(n)$ time and requires n random bits. Now we claim that this algorithm is indeed a 1/2-approximation algorithm for the Max-Cut Problem.

Claim 4.2. *Given $G = (V, E)$ and the result S from the algorithm above, $\mathbb{E}[|cut(S)|] \geq 1/2|cut(S^*)|$, where S^* is the optimal solution for G.*

Proof

Let

$$\mathbb{1}_{e=(u,v)} = \begin{cases} 1 & \text{if } e \in \text{cut}(S) \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

$|\text{cut}(S)| = \sum_{e \in E} \mathbb{1}_e$

For each individual $\mathbb{1}_e$, $\mathbb{E}[\mathbb{1}_e] = \Pr[\text{Exactly one of } u, v \in S] = \frac{1}{2} * \frac{1}{2} + \frac{1}{2} * \frac{1}{2} = \frac{1}{2}$

Therefore, $\mathbb{E}[|\text{cut}(S)|] = \sum_{e \in E} \mathbb{E}[\mathbb{1}_e] = \frac{|E|}{2}$ by linearity of expectation. □

A natural question to ask is if one could remove the randomness used in the algorithm to get a deterministic 1/2-approximation algorithm. Notice that $\mathbb{E}[\mathbb{1}_e]$ is equivalent to $\mathbb{E}_{r \sim \{0,1\}^n}[\text{size}(G, r)]$ if we add the randomness r into the equation, where the $\text{size}(G, r) = |\text{cut}(S)|$. Therefore, we know that $\exists r \in \{0, 1\}^n$ such that $\text{size}(G, r) \geq \frac{|E|}{2}$.

One naive way to derandomize is algorithm is to sample all possible r. However, this is the same as trying all the subsets of V and compare their cuts. This is not helpful at all since it runs in $O(2^n)$ (and also finds the optimal set). Then the question becomes, how can we find a small subset of the r's to search over.

We can expand the equation we used for $\mathbb{E}[\mathbb{1}_e]$ a little bit. Say we sample $X \sim D \to (x_1, x_2, ..., x_n)$ where D is some arbitrary distribution. Follow the algorithm we have above, we let node $v_i \in S$ iff $x_i = 1$. Then $\mathbb{E}[\mathbb{1}_e] = \Pr[(x_u = 0 \wedge x_v = 1) \vee (x_u = 1 \wedge x_v = 0)]$. Notice how we are only examine the relationships between two literals instead of all of them. So we actually did not use the "full power" of complete randomness. So our derandomization is the following:

Algorithm 4.3. *Given input $G = (V, E)$, we use $l = log(n+1)$ random bits to sample $D \sim \{0,1\}^n$ which is pairwise independent (using the algorithm we discussed in lecture 3). Then we sample $X \sim D \to (x_1, x_2, ..., x_n)$ and let node $v_i \in S$ iff $x_i = 1$. There're a total of 2^l possible combinations of random bits, and we brute force all of them and take the one set S with the largest cut.*

Claim 4.4. *We claim that for this derandomized algorithm $\mathbb{E}_{X \sim D}[|cut(S)|] \geq \frac{|E|}{2}$.*

Proof

The proof is basically the same as the proof we have for the randomized algorithm.

$\mathbb{E}[\mathbb{1}_e] = \Pr[(x_u = 0 \wedge x_v = 1) \vee (x_u = 1 \wedge x_v = 0)]$

$= \Pr[(x_u = 0 \wedge x_v = 1)] + \Pr[(x_u = 1 \wedge x_v = 0)]$

$= \frac{1}{2} * \frac{1}{2} + \frac{1}{2} * \frac{1}{2} = \frac{1}{2}$ by the definition of pairwise independence.

$\mathbb{E}[|\text{cut}(S)|] = \sum_{e \in E} \mathbb{E}[\mathbb{1}_e] = \frac{|E|}{2}$. And since we pick the largest one from all 2^l samples, we know $|\text{size}(S)| \geq \frac{|E|}{2}$(otherwise the mean cannot reach $\frac{|E|}{2}$).□

Using the properties of pairwise independence, instead of brute force all 2^n subsets, we only need to brute force over 2^l choices of random bits. Since $l = log(n+1)$, we reduce the number of times from $O(2^n)$ to $n+1$.

4.2 K-wise Independence

As we saw above, in many cases, one can use limited independence instead of using completely uniform bits. We now give a randomness efficient way of generating k-wise independent distributions (defined below).

Definition 4.5. *$X_1, X_2, ..., X_N$ are k-wise independent random variables if for each $X_i \sim [M]$, $\forall$distinct $i_1, i_2, ..., i_k$, $Pr[X_{i1} = x_{i1} \wedge X_{i2} = x_{i2} \wedge ... \wedge X_{ik} = x_{ik}] = \frac{1}{M^k}$.*

The construction for k-wise independent distribution is the following:

Let $N = 2^n$ (for simplicity let $M = N$), $\mathbb{F} = \mathbb{F}_N$

- Sample $\vec{a} = (a_0, a_1, ..., a_{k-1}) \sim \mathbb{F}$ (this uses $k * log(N)$ bits)
- Define univariate polynomial $P_{\vec{a}}(x) = \sum_{i=0}^{k-1} a_i x^i$ and output distribution $D \sim P_{\vec{a}}(0), P_{\vec{a}}(1), ..., P_{\vec{a}}(N-1)$, where we evaluate all elements in $\mathbb{F}$. Each $P_{\vec{a}}(i)$ represents the random variable X_i.

Note how this construction is a generalization to the pairwise independence. When $k = 2$, we get exactly $P_{\vec{a}}(x) = ax + b$.

Proof

$X_i = P_{\vec{a}}(i)$, let $i_1, i_2, ..., i_k$ be distinct field elements. We want to show that $\Pr[X_{i1} = x_{b1} \wedge X_{b2} = x_{b2} \wedge ... \wedge X_{bk} = x_{bk}] = \frac{1}{M^k}$, where b_i's are arbitrary field elements.

$\Pr[X_{i1} = x_{b1} \wedge X_{b2} = x_{b2} \wedge ... \wedge X_{bk} = x_{bk}]$

$$= \Pr[\ \forall_{j=1}^{k} P_{\vec{a}}(i_j) = b_j]$$

$$= \Pr\left[\begin{pmatrix} 1 & i_1 & i_1^2 & ... & i_1^{k-1} \\ 1 & i_2 & i_2^2 & ... & i_2^{k-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & i_k & i_k^2 & ... & i_k^{k-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{k-1} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{pmatrix}\right]$$

The $k \times k$ matrix (say M) here is called the Vandermonde matrix, and it's determinant is $\det(M) = \prod_{1 \le i \le j \le n}(x_j - x_i)$, which is non-zero if all x_i's are distinct. Since all i_j's are distinct from our assumption, $\det(M) \neq 0$, and M is an invertible matrix. Therefore, $M\vec{a} = \vec{b}$ has a unique solution here. With the similar logic we used to prove pairwise independence, $\Pr[X_{i1} = x_{b1} \wedge X_{b2} = x_{b2} \wedge ... \wedge X_{bk} = x_{bk}] = \frac{1}{M^k}$. $\square$

For this construction, under the domain of $[N]^N$, note that the support size is just $\le N^k$. Thus, It significantly reduces the randomness we use. I k is constant, N^k is within polynomial level. Then we may want to know how optimal is this construction. That is to say, can we do better than N^k?

Claim 4.6. *For any k-wise independent distribution D on $\{0,1\}^n$, $|support(D)| \ge n^{\lfloor \frac{k}{2} \rfloor}$*

We will supply a proof in the next lecture.