

## Lecture 21: 27-10-2022

Scribe: Amit Rajaraman

Lecturer: Rohit Gurjar

**Lemma 21.1** (Isolation Lemma). *Let  $E$  be a set of  $m$  elements and  $\mathcal{S} \subseteq 2^E$  be an arbitrary family of subsets of  $E$ . Independently and uniformly randomly assign to each element of  $E$  a weight in  $\{1, 2, \dots, N\}$ . Then,*

$$\Pr[\mathcal{S} \text{ has a unique minimum weight set}] \geq 1 - \frac{m}{N},$$

where the weight of a set is the sum of the weights of the elements in it.

We get the desired result in the context of matchings on setting  $E$  to be the set of edges and  $\mathcal{S}$  to be the collection of perfect matchings.

*Proof.* Let  $E = \{e_1, \dots, e_m\}$ . Split  $\mathcal{S}$  into two parts  $\mathcal{S}_0, \mathcal{S}_1$ , where  $\mathcal{S}_0 = \{T \in \mathcal{S} : e_1 \in T\}$  and  $\mathcal{S}_1 = \mathcal{S} \setminus \mathcal{S}_0$ . Let us look at the event  $\mathcal{E}$  that there is both a minimum weight set that contains  $e_1$  and a minimum weight set that does not contain  $e_1$ . This means that the weights of the minimum weight sets in  $\mathcal{S}_0$  and  $\mathcal{S}_1$  are equal.

What happens if we fix the weights of all elements other than  $e_1$ ? The minimum weight in  $\mathcal{S}_1$  is determined, and the minimum weight in  $\mathcal{S}_0$  is just equal to some fixed quantity plus the weight of  $e_1$ . In particular, there is at most one value of  $w(e_1)$  such that the two minimum weights are equal. Therefore,  $\Pr[\mathcal{E}] \leq 1/N$ . In general, taking the union bound, we have

$$\begin{aligned} & \Pr[\text{there exist two minimum weight sets}] \\ &= \Pr \left[ \bigcup_{i \in [m]} \text{there exist minimum weight sets containing } e_i \text{ and not containing } e_i \right] \leq \frac{m}{N}. \end{aligned}$$

□

Later, it was proved that the above probability bound can be improved to  $(1 - \frac{1}{N})^m$ .

The isolation lemma has several surprising applications, for example that **UNIQUE-SAT**<sup>1</sup> is NP-hard.

We next look at derandomization. We cannot derandomize the isolation lemma in all its generality, but we can for specific families that have some structure.

For example, this is very easy for spanning trees and it suffices to assign distinct weights to all edges. Our main goal is that of derandomizing isolation lemma for *bipartite perfect matching*. We will only be able to derandomize it to  $O(\log^2 n)$  random bits unfortunately, which is equivalent to giving  $n^{O(\log n)}$  weight assignments with the assurance that one of them gives a minimum weight matching.

The high-level view of the proof is the following.

The weight construction is done in  $\log n$  rounds. We start off with some huge (exponentially large) family of perfect matchings. We then come up with some weight function such that the set of perfect matchings of minimum weight is comparatively smaller. We then come up with another weight function (with about  $\log n$  bits) to break ties among these minimum weight perfect matchings and make the set even smaller. Further, we ensure that the older non-minimum weight matchings do not suddenly enter this family by appending the bits of the new weight function to the bits of the previous weight function. After repeating this, we finally arrive at a weight configuration such that there is only one minimum weight matching. Each of these bit sequences we append are  $\log n$  bits, and because there are  $\log n$  rounds we end at  $\log^2 n$  bits in all.

<sup>1</sup>This is SAT, except that we know that if there is a satisfying assignment, it is unique.

As before, let the edges be  $e_1, \dots, e_m$ .

For starters, observe that if  $w(e_i) = 2^i$  for all  $i$ , then all subsets have distinct weights.

Let  $M_1, M_2$  be two minimum weight perfect matchings. Observe that  $M_1 \cup M_2$  is a union of cycles (and possibly isolated edges contained in both  $M_1, M_2$ ). Further, each cycle in  $M_1 \cup M_2$  has zero “alternating weight”. This is the difference of the sum of all “odd” edges in the cycle and the sum of all “even” edges in the cycle. Indeed, if we instead had that the  $M_1$  sum was greater than the  $M_2$  sum, we could switch out the edges in the cycle in  $M_1$  for edges in the cycle in  $M_2$  to get a matching of weight strictly less than that of  $M_1$ , yielding a contradiction. Below we have a much stronger version of this fact.

**Lemma 21.2.** *Let  $E' \subseteq E$  be the union of all minimum weight perfect matchings. Then, each cycle in  $G = (V, E')$  has zero alternating weight.*

**Corollary 21.3.** *If  $w$  is a weight assignment such that a cycle  $C$  has nonzero alternating weight, then the union of minimum weight perfect weight matchings (with respect to  $w$ ) does not contain  $C$ .*

The above corollary is the key idea. For a suitable weight assignment on a cycle, we can get rid of at least one edge in the cycle, and this ensures that all matchings containing that edge are rid of. Our goal then is to maximize the number of edge-disjoint cycles in the graph.

Given a cycle  $C$  and a weight assignment  $w$ , let  $aw(C)$  be the alternating weight of  $C$  under  $w$ .

**Proposition 21.4.** *Let  $C_1, \dots, C_k$  be an arbitrary collection of cycles. Then, for some  $j \in \{1, 2, \dots, m^2k\}$ , the weight function defined by  $w(e_i) = 2^i \pmod{j}$  for each  $i$  assigns a nonzero alternating weight to every cycle  $C_r$ .*

*Proof.* We would like to show that for some  $j$ ,  $j$  is not a factor of the product  $aw(C_1)aw(C_1) \cdots aw(C_k)$ . This product is at most  $2^{m^2k}$ . Recalling that the lcm of the first  $n$  numbers is greater than  $2^n$  for sufficiently large  $n$ , we have that  $2^{m^2k}$  is less than the lcm of  $[m^2k]$ , so there is some number in  $[m^2k]$  that does not divide the product.  $\square$

Note that the list of weight assignments we give as above does not require knowledge of which cycles we are working with. That is, if we have polynomially many cycles, we can give a polynomially large list of weight assignments with the guarantee that one of these assignments removes all the cycles.