# Lecture 2: Lovász Local Lemma + Beck's Algorithm

*Lecturer: Ronitt Rubinfeld* *Scribe: Vivian Qian*

# 1 Lovász Local Lemma

## 1.1 Motivation

We want to argue that there is a nonzero probability that no "bad events" occur. If $A_1, A_2, ..., A_n$ are "bad events", what is the probability that none of them occur?

**Usual Way** If we assume nothing about $A_i$'s wrt independence, we can use union bound:

$$\Pr[\cup A_i] \leq \sum_i \Pr[A_i]$$

If each $A_i$ occurs with probability $p$, and $p < \frac{1}{n}$, then $\Pr[\cup \overline{A_i}] > 0$.

**Independence** If we assume $A_i$'s are independent and nontrivial (e.g. $\Pr[A_i] < 1$), then:

$$\begin{aligned} \Pr[\cup A_i] &\leq 1 - \Pr[\cap \overline{A_i}] \\ &= 1 - \prod_i \Pr[\overline{A_i}] < 1 \end{aligned}$$

Therefore, there is always a nonzero probability that no $A_i$ occurs under these assumptions.

What if $A_i$'s have only "some" independence?

## 1.2 Lovász Local Lemma

**Definition 1.** *$A$ is **independent** from $B_1, ..., B_k$ if $\forall J \in [k]$ where $J \neq \emptyset$:*

$$\Pr[A \cap \bigcap_{j \in J} B_j = \Pr[A] * \Pr[\bigcap_{j \in J} B_j]$$

**Definition 2.** *Given events $A_1, ..., A_n$, $D = (V, E)$ with $V = [n]$ is a **dependency graph** of $A_1, ..., A_n$ if each $A_i$ is independent of all $A_j$ that aren't its neighbors in $D$.*

**Theorem 3** (symmetric Lovász Local Lemma)**.** *Let $A_1, ..., A_n$ be events s.t. $\Pr[A_i] \leq p \forall i$, with dependency graph $D$ of degree $\leq d$. If $ep(d+1) \leq 1$, then $\Pr[\bigcap_{i=1}^n \overline{A_i}] > 0$.*

Notice that this has no dependency on the number of events, $n$. If the degree $d$ is small, this is better than the union bound.

## 1.3 Two Coloring Application

**Theorem 4.** *Let $S_1, ...S_m \in X$ where $|S_i| = l$ and each $S_i$ intersects with **at most** $d$ other $S_j$'s. If $e(d+1) \leq 2^{l-1}$, then a 2-coloring exists s.t. each $S_i$ is not monochromatic.*

*Proof.* Randomly color each element of $X$ red or blue. Let $A_i$ be the event that $S_i$ is monochromatic. The probability $p$ that $A_i$ occurs is the probability that all elements are red or blue, which is $\frac{1}{2^{l-1}}$. Since each $S_i$ intersects with at most $d$ other $S_j$'s, and $A_i$ is only dependent on $A_j$ if their intersection is nonempty, the dependency graph $D$ of $A_1, ..., A_n$ has degree $d$. By Lovász Local Lemma, since

$$ep(d+1) = e * 2^{-(l-1)}(d+1) \leq 1,$$

there exists a 2-coloring such that no $S_i$ is monochromatic. □

**Second Application** Given a CNF formula with $l$ variables in each clause, with each variable in at most $k$ clauses, if

$$\frac{e(lk+1)}{2^{l-1}} \leq 1,$$

there exists a satisfying assignment.

# 2 Moser-Tardos Algorithm

**Theorem 5.** *Let $S_1, ..., S_m \in X$ be sets with $|S_i| = l$ where each $S_i$ intersects with at most $d$ other $S_j$'s. If $c * e(d+1) \leq 2^{l-1}$, for some constant $c > 1$, then a 2-coloring exists such that each $S_i$ is not monochromatic.*

## 2.1 Moser-Tardos Algorithm

1. Randomly assign each element of $X$ to red or blue.
2. While there exists a monochromatic set:
   - Choose an arbitrary monochromatic set $S_i$
   - Randomly reassign colors of all elements in $S_i$.

# 3 Beck's Algorithm

**Stronger Assumptions** Let $D = d^4$. Assume $l$ is **constant** and that $16D(d+1) < 2^l$.

## 3.1 Beck's Algorithm

**Algorithm 1:** Beck's Algorithm

```
Given S_1, ..., S_m ∈ X;
First Pass;
for each element j ∈ X do
    if j is "frozen" then
        do nothing;
    else
        pick color ∈ red, blue via coin flip;
        consider all S_j containing j
        if S_j has l_1 points the same color and no points in the other color then
            S_j becomes dangerous;
            all uncolored points in S_j are "frozen";
        else
            pick color ∈ red, blue via coin flip;
        end
    end
end
if S_i is not yet 2 colored, then it "survives";

Second Pass;
Use brute force to find coloring of surviving S_i's.
```

## 3.2 Analysis

**Question** How can we prove correctness and runtime of Beck's Algorithm?

We consider a single $S_i$. The probability that it survives is at least as likely as the probability that $S_i$ becomes dangerous. This is because a set $S_i$ can survive if its intersecting elements are frozen by neighboring sets:

$$\begin{aligned}\Pr[S_i \text{ survives}] &\geq \Pr[S_i \text{ is dangerous}] \\ &= \frac{2}{2^{l_1}} && \text{// P(all red) + P(all blue)} \\ &= 2^{1-l_1}\end{aligned}$$

*

The probability $S_i$ is dangerous is exactly if all $l_1$ elements are red or blue.

We now consider how two different sets $S_i$ and $S_j$ are related. The survival of $S_i$ and $S_j$ is not necessarily independent. Consider the case where $S_i \cap S_j \neq \emptyset$. If the intersecting points are frozen by $S_i$, the probability that $S_j$ survives is higher. We can extend this logic even in cases where they do not directly overlap.

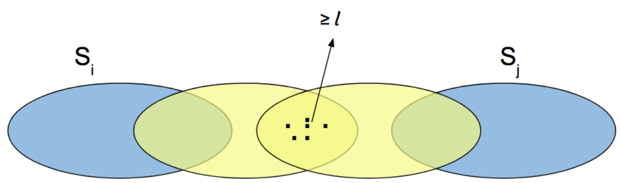


**Figure 1**: $S_i$ and $S_j$ dependent

In Figure 1, if the shown $\geq l$ points are monochromatic, the middle two sets will become dangerous. Since the middle two overlap with either $S_i$ or $S_j$, this increases the probability that either survives. Therefore, $S_i$ and $S_j$ are not independent.

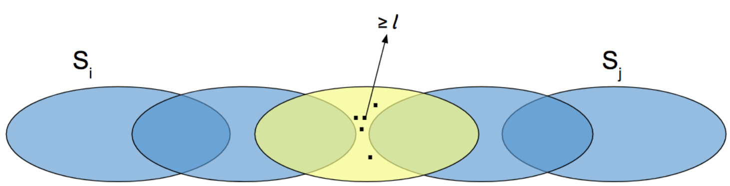


**Figure 2**: $S_i$ and $S_j$ independent

However, if we consider the case in Figure 2, where $S_i$ and $S_j$ are a distance 4 apart, the middle set becoming dangerous does not affect either $S_i$ or $S_j$. We conclude that if the distance between $S_i$ and $S_j$ is $\geq 4$, they survive independently of each other.

We will construct an useful graph $G$ to prove correctness.

$$\begin{aligned} G &\leftarrow \text{nodes } V \leftarrow [m] \text{ (each node is a set } S_i) \\ &\leftarrow \text{edges } (i,j) \in E \text{ iff } S_i \cap S_j \neq \emptyset \end{aligned}$$

Observe by LLL, a solution exists for this graph $G$. We claim the following:

**Claim** After the first pass, with high probability, all surviving nodes of $G$ form connected components of size $O(\log m \text{ poly}(d))$.

**Corollary** If the above claim is true, then if $l$ is constant, the second pass only needs to brute force $O(2^{lc \log m}) = O(m^{lc})$, which is polynomial in $m$.

*Proof.* Consider the biggest tree $T \in C$ such that $C$ is a component that survives and:

1. all nodes in $T$ are a distance $\geq 4$ in $G$
2. if the nodes in $T$ of distance $= 4$ are connected in $G^4$ then $T$ is connected

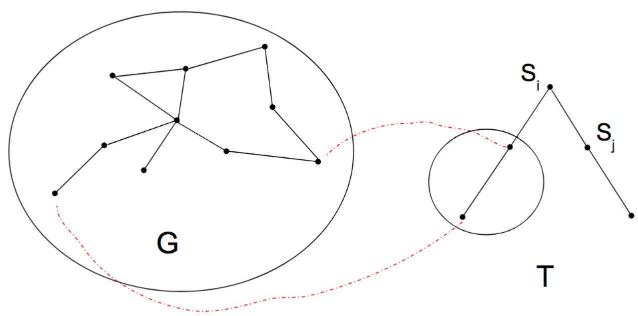


**Figure 3**: Nodes in T $\geq$ 4 distance in G

If we pick $T$ greedily, there exists a $T$ whose size is $\geq \frac{|C|}{d^3}$. If $C$ survives, then $T$ also survives since $T \subseteq C$. Thus, $\Pr[T \text{ survives}] \geq \Pr[C \text{ survives}]$. Looking at Fig. 1 and 2, for a $S_i \in T$ to survive, it must either:

- $S_i$ is dangerous
- $S_i$ is next to a dangerous $S_j$ that froze its elements

However, since in $T$ all elements are at least distance 4, $S_i \cap S_j = \emptyset$. For each $S_i \in T$, we pick a neighbor $S_{i'}$ in $(d+1)^k$ possible ways where $k = |T|$. Given all $S_{i'}$ are disjoint, the probability that all $k$ become dangerous is:

$$\Pr[k \text{ nodes } S_{i'} \text{ become dangerous }] \leq (2^{(1-l)})^k$$

Using union bound, the probability that all $S_i$ survive:

$$\Pr[\text{all } S_{i'} \text{ survive}] \leq (d+1)^k \cdot 2^{k(1-l)}$$

We will now show that no such large tree survives after the first pass. The number of trees of size $u$ that could exist in $G$ is $\leq m(d^4)^u$. There are $m$ different choices for the root of tree $T$ and $d^4$ choices for each subsequent node, as the next node is distance $\geq 4$ away in $G$. Therefore,

$$\begin{aligned} \text{E[ trees of size u that survive]} &\leq m(d^4)^u \cdot (d+1)^u \cdot 2^{u*(1-l)} \\ &= m(d^4(d+1) \cdot 2^{(1-l)})^u \end{aligned}$$

Because of our earlier assumption that $16d^4(d+1) < 2^l$, we can simplify further:

$$\begin{aligned} \text{E[ trees of size u that survive]} &\leq m(2^l \cdot 2^{1-l_1})^u \\ &= m(2^u) \end{aligned}$$

If $u \geq \Omega(\log m)$, then the expected number of trees is $o(1)$. Thus Beck's algorithm runs in polynomial time in $m$. $\square$