CS787: Advanced Algorithms	**Scribe:** Pan Wu / Sapan Gupta
Lecture 26: Algorithmic LLL	**Date:** 05/02/2019

26.1 Lovász local lemma and its algorithmic (constructive) version

26.1.1 Symmetric LLL

Let $\varepsilon_1, ..., \varepsilon_n$ be events with a dependency graph of degree at most d.
Suppose $\forall i$, $P[\varepsilon_i] \leq \rho$, and $e\rho(d+1) \leq 1$, where $e = 2.71$
Then,

$$P[\bigcup_{i=1}^{n} \varepsilon_i] \leq 1 - (\frac{d}{d+1})^n < 1$$

26.1.2 General LLL

Let $\varepsilon_1, ..., \varepsilon_n$ be events and let Γ_i denote the neighbors of ε_i in their dependency graph.
Suppose there exists numbers $x_i \in (0,1)$ such that $\forall i$, $P[\varepsilon_i] \leq x_i \prod_{j \in \Gamma_i}(1-x_j)$
Then,

$$p[\bigcap_{i=1}^{n} \tilde{\varepsilon_i}] \geq \prod_{i=1}^{n}(1-x_i) > 0$$

Dependency graph between events:
- n vertices
- Place an edge between i and j, if i depends on j

For $\Gamma_i = neighbors\ of\ i$,
$\forall i, \forall S \subseteq [n] \backslash \Gamma_i^+$,
$\Gamma_i^+ = \Gamma_i \cup \{i\}$,

$$P[\varepsilon_i | \bigcup_{j \in S} \varepsilon_j] = P[\varepsilon_i]$$

26.1.3 *GLL => SLL*

Set $x_i = \frac{1}{d+1}, \forall i$
For any i,

$$x_i \prod_{j \in \Gamma_i}(1-x_j) \geq \frac{1}{d+1}(1-\frac{1}{d+1})^d \geq \frac{1}{d+1} \cdot \frac{1}{e} \geq p \geq P[\varepsilon_i]$$

Thus, SLLL is proved using GLLL.

Special case:
Suppose that $\forall i$, $\sum_{j\in\Gamma_i} P[\varepsilon_j] \leq \frac{1}{4}$.
Then,

$$P[\bigcap_{i=1}^{n} \tilde{\varepsilon_i}] \geq \prod_{i=1}^{n}(1-p[\varepsilon_i]) > 0$$

Proof GLLL:
Set $x_i = 2P[\varepsilon_i]$

$$x_i \prod_{j\in\Gamma_i}(1-x_j) \geq 2P[\varepsilon_i](1-\sum_{j\in\Gamma_i} x_j) \geq 2P[\varepsilon_i](1-\frac{1}{2}) = P[\varepsilon_i]$$

26.1.4 Proof of GLLL (Induction on size $|S|$)

Inductive hypothesis:

$$P[\bigcap_{i\in S} \tilde{\varepsilon_i}] \geq \prod_{i\in S}(1-x_i) \quad and \quad P[\varepsilon_i | \bigcap_{j\in S} \tilde{\varepsilon_j}] \leq x_j$$

$$\iff P[\varepsilon_i | \bigcap_{j\in S} \tilde{\varepsilon_j}] \geq 1-x_i$$

Consider the set $S = \{1,2,...,k,k+1\}$

$$P[\bigcap_{i=1}^{k+1} \tilde{\varepsilon_i}] = P[\bigcap_{i=1}^{k} \tilde{\varepsilon_i}].P[\overline{\varepsilon}_{k+1} | \bigcap_{i=1}^{k} \tilde{\varepsilon_i}] \geq \prod_{i=1}^{k}(1-x_i)(1-x_{k+1})$$

$$P[\varepsilon_i | \bigcap_{j\in S} \tilde{\varepsilon_j}] = \frac{P[\varepsilon \quad and \quad \bigcap_{j\in\Gamma_i\cap S} \tilde{\varepsilon_j} | \bigcap_{j\in S\setminus\Gamma_i} \overline{\varepsilon}_j]}{P[\bigcap_{j\in\Gamma_i\cap S} \tilde{\varepsilon_j} | \bigcap_{j\in S\setminus\Gamma_i} \overline{\varepsilon}_j]}$$

$$\leq \frac{P[\varepsilon_i | \bigcap_{j\in S\setminus\Gamma_i} \overline{\varepsilon}_j]}{P[\overline{\varepsilon}_1 | -]P[\overline{\varepsilon}_2 | \overline{\varepsilon}_1 \bigcap -]P[\overline{\varepsilon}_3 | \overline{\varepsilon}_1 \bigcap \overline{\varepsilon}_2 \bigcap -]}$$

$$\leq \frac{P[\varepsilon_i]}{(1-x_1)(1-x_2)(1-x_3)...}$$

$$\leq \frac{x_i \prod_{j\in\Gamma_i}(1-x_j)}{\prod_{j\in\Gamma_i \bigcap S}(1-x_j)} \leq x_i$$

26.1.5 k-SAT

Given any k-CNF formula where every variable belongs to at most $\frac{2^k}{ek}$ clauses, the formula is satisfiable. (every clause has exactly k distinct literals)

[Moser'09]

Algorithm:
- Pick a uniform random assignment while $\exists$ unsat clause C, FIX(C).

FIX(C)
- Pick a uniform random assignment for variable in C.
- While $\exists$ unsatisfied clause D that shares a variable with C, run FIX(D).

If degree of $C \leq \frac{2^k}{e}$, algorithm is going to terminate quickly. Every clause that is touched in the recursive FIX calls, becomes satisfied. Everything else remains untouched. If algorithm encounters loops, it may not terminate.