1. Moser-Tardos Construction for Lovasz Local Lemma (final part)

Recall from previous lectures that $E_1, E_2, \ldots, E_n$ is the set of random events and $T_{E_i}$ is the set of proper witness trees with root labeled $E_i$.

From the previous lecture, we will be done once we prove the following lemma.

**Lemma 1.** For any $E_i$

$$\sum_{T \in T_{E_i}} \prod_{u \in T} P[E_{i_u}] \leq \frac{x_i}{1 - x_i}$$

**Proof.** Consider the following random Galton-Watson process of generating proper witness trees with root labeled $E_i$.

Initially let the only vertex be the root labeled by $E_i$

**loop** while there is a variable to add

- Consider each vertex $u$ at depth $t - 1$. $u$ is labeled by $E_{i_u}$.
- Consider each $j \in \Gamma_{i_u}^+$
  - Add a child to $E_{i_u}$ labeled $E_j$ with probability $x_j$.
  - Don’t add it with probability $(1 - x_j)$.

**end loop**

Fix a proper witness tree $T \in T_{E_i}$. We calculate the probability that the above Galton-Watson process generates the tree $T$.

Let $u$ be a vertex in $T$. Let $W_u = \{j \in \Gamma_{i_u}^+ \mid E_j \text{ does not label any child of } u\}$.

If $u$ has already been correctly generated in $T$ by the process, then the probability of adding the “correct” children to generate $T$ is

$$\left( \prod_{j \in W_u} (1 - x_j) \right) \left( \prod_{j \in \Gamma_{i_u} \setminus W_u} x_j \right)$$
Taking product of this over all $u$ and regrouping gives the probability of the Galton-Watson process generating $T$.

$$P_T = \frac{1}{x_i} \prod_{u \in T} \left( x_i \prod_{j \in W_u} (1 - x_j) \right)$$

$$= \frac{1 - x_i}{x_i} \prod_{u \in T} \left( \frac{x_{i_u}}{1 - x_{i_u}} \prod_{j \in \Gamma_{i_u}^+} (1 - x_j) \right)$$

$$= \frac{1 - x_i}{x_i} \prod_{u \in T} \left( x_{i_u} \prod_{j \in \Gamma_{i_u}} (1 - x_j) \right)$$

$$= \frac{1 - x_i}{x_i} \prod_{u \in T} x'_{i_u}$$

We have

$$\sum_{T \in T_{E_i}} P_T \leq 1$$

Recalling that $P[E_{i_u}] \leq x'_i = x_i \cdot \prod_{j \in \Gamma_i} (1 - x_j)$, this implies

$$\sum_{T \in T_{E_i}} \prod_{u \in T} P[E_{i_u}] \leq \sum_{T \in T_{E_i}} \prod_{u \in T} x'_{i_u} \leq \frac{x_i}{1 - x_i}$$

---

2. PPSZ algorithm for Satisfiability (Part I)

We covered the WalkSat algorithm for finding a truth assignment for an instance of $k$-SAT. We present another algorithm due to Paturi, Pudlák, Saks and Zane [2] that improves on the running time.

An improvement was given by Hertli [1] in 2011. The following theorem states the existence of such an algorithm.

**Theorem 1.** Let $S_k = \int_0^1 \frac{t^{i/k} - t}{1-t}.dt$ and $s_k = 2^{S_k}$. Then there is a randomized algorithm that, given a satisfiable instance of $k$-SAT, finds a satisfying assignment in expected time “close to” $(s_k)^n = 2^{S_k n}$.

The following table gives a comparison of the exponent in the running time of PPSZ with WalkSat.
We first define some preliminaries and then proceed with the description of the algorithm.

**Definition 1.** For a set of clauses $\Gamma$ and a literal $x$, we write $\Gamma \models x$, (read $\Gamma$ implies $x$), if all truth assignments that satisfy $\Gamma$ also set $x$ to True.

**Definition 2.** Let $D \geq 1$, then $\Gamma \models D \cdot x$, if and only if for some $\Gamma_0 \subseteq \Gamma$, $|\Gamma_0| \leq D$, $\Gamma_0 \models x$.

**Observation:** For constants $k, D$ and $\Gamma$ an instance of $k$-SAT, there is a polynomial time algorithm to check if $\Gamma \models D \cdot x$.

The observation is true because we can use a brute force algorithm to check for all $\Gamma_0 \subseteq \Gamma$ with $|\Gamma_0| \leq D$ whether $\Gamma_0 \models x$. Since $\Gamma_0$ has at most $D \cdot k$ many distinct variables, there are only $2^{D \cdot k}$ many truth assignments, which is constantly many.

Let $\Gamma$ be a set of clauses over the variables $x_1, \ldots, x_n$. Let $\alpha$ be a partial truth assignment; i.e., domain($\alpha$) $\subseteq \{x_1, \ldots, x_n\}$ and range($\alpha$) $= \{T, F\}$.

**Definition 3.** $\Gamma^\alpha$ or $\Gamma|_\alpha$ is the set of clauses obtained by

1. Removing all clauses in $\Gamma$ that contain an $x$ s.t. $\alpha(x) = T$.
2. Erasing from $\Gamma$ any literal $x$ s.t. $\alpha(x) = F$.

We now present the PPSZ algorithm

**Input:** $\Gamma, D$
**Output:** A truth assignment $\alpha$

Initialize $\alpha$ = empty partial truth assignment

while True do
    if $\alpha$ satisfies $\Gamma$ then
        return $\alpha$
    if domain($\alpha$) $\subseteq \{x_1, \ldots, x_n\}$ then
        return FAILURE
    if $\exists x \mid \Gamma|_\alpha \models \Gamma \models_D \cdot x, x \notin \text{domain}(\alpha)$ then
        set $\alpha(x) = T$
    else
        pick $x \in \{x_1, \ldots, x_n\}$ at random and set $\alpha(x) = \{T, F\}$ at random.
end

**Algorithm 1:** The PPSZ algorithm
The following theorem will be proved in the next two lectures.

**Theorem 2.** For any $s'_k > s_k = 2^{S_k}$, $\exists D$, sufficiently large such that $PPSZ(\Gamma, D)$ finds a satisfying assignment with probability greater than $(s'_k)^{-n}$.

Iterating $PPSZ(\Gamma, D)$ gives the algorithm for the PPSZ/Hertli Theorem, with expected number of iteration to find a satisfying assignment $\leq (s'_k)^n$, assuming $\Gamma$ is satisfiable.

**References**
