Lecture Outline:

- General Lovász Local Lemma
- Two Simple Applications of the LLL
- Packet Routing

In this lecture, we prove the general form of the Lovász Local Lemma, and see two applications of the lemma: hypergraph coloring and finding satisfying assignments for Boolean CNF formulas. We then begin studying packet routing, another area in which we will apply the lemma.

1 Lovász Local Lemma

In the last lecture, we proved the symmetric form of the Lovász Local Lemma (LLL). Today we will prove the general form. Recall that the LLL is used to show that, for a set of "bad" events over a sample space, there exists a point in the sample space outside of all of the events, under certain conditions on the dependence and probabilities of the events.

Theorem 1 (General LLL). *Let $E_1, \ldots, E_n$ be a collection of events over some sample space. For each E_i, let $S_i = \{j \in [n] \setminus \{i\} \mid E_i \text{ is dependent on } E_j\}$. If there exist $x_1, \ldots, x_n \in (0, 1]$ such that*

$$\forall i: \ \Pr[E_i] \leq x_i \prod_{j \in S_i} (1 - x_j)$$

then $\Pr\left[\bigwedge_i \overline{E_i}\right] \geq \prod_i (1 - x_i)$.

Before proving this, it is useful to note that the symmetric version (restated here for convienience) follows easily from the general.

Theorem 2 (Symmetric LLL). *Let $E_1, \ldots, E_n$ be a collection of events such that $\forall i : Pr[E_i] \leq p$. Suppose further that each event is dependent on at most d other events, and that $e \cdot p \cdot (d + 1) \leq 1$. Then,* $\Pr\left[\bigwedge_i \overline{E_i}\,\right] > 0$.

Proof of Theorem 2 assuming Theorem 1. To apply Theorem 1, choose $x_i = 1/(d + 1)$ for each i. Then for each event, we indeed have

$$\Pr[E_i] \leq p \leq \frac{1}{e(d+1)} \leq \frac{1}{d+1} \cdot \left(1 - \frac{1}{d+1}\right)^d \leq x_i \prod_{j \in S_i} (1 - x_j)$$

and so $\Pr\left[\bigwedge_i \overline{E_i}\right] \geq \prod_i (1 - x_i) > 0$. □

The proof of the general case is nearly identical to the proof of the symmetric case that we saw in the last lecture. As before, most of the work is done by a lemma.

Lemma 1. *Assume the conditions of Theorem 1. Then,* $\forall\, T \subseteq \{E_1, \ldots, E_n\}$ *and* $\forall\, E_i$*:*

$$\Pr\left[E_i \;\middle|\; \bigwedge_{E_j \in T} \overline{E_j}\right] \leq x_i$$

Proof. The proof is by induction on the size of T. For the base case $|T| = 0$, the lemma holds because $\Pr[E_i] \leq x_i \prod_{j \in S_i}(1 - x_j) < x_i$. Assume it holds on all sets up to size $k-1$. Fix T to be a set of size k and fix an event E_i. Let $S \subseteq T$ be defined such that E_i depends on all events in S and no events in $T \setminus S$; formally, $S = T \cap \{E_j \mid j \in S_i\}$. Assume that $|S| \geq 1$ (i.e. that E_i depends on some event in T); otherwise we reduce to the base case. The chain rule for probabilities, and the fact that $S \cup (T \setminus S) = T$, gives

$$\Pr\left[E_i \;\wedge \bigwedge_{E_j \in S} \overline{E_j} \;\middle|\; \bigwedge_{E_\ell \in T \setminus S} \overline{E_\ell}\right] = \Pr\left[\bigwedge_{E_j \in S} \overline{E_j} \;\middle|\; \bigwedge_{E_\ell \in T \setminus S} \overline{E_\ell}\right] \cdot \Pr\left[E_i \;\middle|\; \bigwedge_{E_j \in T} \overline{E_j}\right]$$

This can be rewritten as

$$\Pr\left[E_i \;\middle|\; \bigwedge_{E_j \in T} \overline{E_j}\right] = \frac{\Pr\left[E_i \;\wedge\; \bigwedge_{E_j \in S} \overline{E_j} \;\middle|\; \bigwedge_{E_\ell \in T \setminus S} \overline{E_\ell}\right]}{\Pr\left[\bigwedge_{E_j \in S} \overline{E_j} \;\middle|\; \bigwedge_{E_\ell \in T \setminus S} \overline{E_\ell}\right]}$$

Note that the left hand side is the probability we are trying to bound. We will bound each half of the fraction on the right hand side separately. For the numerator, we have

$$\Pr\left[E_i \;\wedge \bigwedge_{E_j \in S} \overline{E_j} \;\middle|\; \bigwedge_{E_\ell \in T \setminus S} \overline{E_\ell}\right] \leq \Pr\left[E_i \;\middle|\; \bigwedge_{E_\ell \in T \setminus S} \overline{E_\ell}\right] \leq x_i \prod_{j \in S_i}(1 - x_j)$$

The second inequality follows from the fact that E_i is independent of the events in $T \setminus S$. To bound the denominator, denote $S = \{E_{j_1}, \ldots, E_{j_{|S|}}\}$. Then,

$$\begin{aligned}
\Pr\left[\bigwedge_{E_j \in S} \overline{E_j} \;\middle|\; \bigwedge_{E_\ell \in T \setminus S} \overline{E_\ell}\right] &= \Pr\left[\overline{E_{j_1}} \mid \bigwedge_{E_\ell \in T \setminus S} \overline{E_\ell}\right] \cdot \Pr\left[\overline{E_{j_2}} \mid \overline{E_{j_1}} \wedge \bigwedge_{E_\ell \in T \setminus S} \overline{E_\ell}\right] \cdot \ldots \\
&\geq (1 - x_{j_1})(1 - x_{j_2}) \cdots (1 - x_{j_{|S|}})
\end{aligned}$$

The inequality follows from the inductive hypothesis, because there are at most $k-1$ events on the r.h.s. of each conditional probability by our assumption that $|S| \geq 1$ and $|T| = k$.

Now, notice that each term in the denominator bound also appears in the numerator bound because of the relationship between S and S_i. Thus, combining the two bounds completes the proof. $\square$

The proof of Theorem 1 is now just an application of the chain rule and Lemma 1.

Proof of Theorem 1.

$$\begin{aligned}
\Pr[\bigwedge_i \overline{E_i}] &= \Pr[\overline{E_1}] \cdot \Pr[\overline{E_2} \mid \overline{E_1}] \cdot \Pr[\overline{E_3} \mid \overline{E_1} \bigwedge \overline{E_2}] \cdot \ldots \cdot \Pr[\overline{E_n} \mid \bigwedge_{i<n} \overline{E_i}] \\
&\geq \prod_i (1 - x_i)
\end{aligned}$$

□

For more on the Lovász Local Lemma (and the first of the two applications we will see next), see the original paper [EL75] and the textbook by Alon & Spencer [AS00].

2 Simple Applications of the Lovász Local Lemma

2.1 Hypergraph Coloring

The Lovász Local Lemma was originally proved to show that a coloring of a certain type of hypergraph exists. In this section we present this result.

Definition 1. A *hypergraph* $G = (V, E)$ consists of a vertex set V and an edge set E, where each edge $f \in E$ is a non-empty subset of V. (Unlike in standard graphs, hypergraph edges may contain more than two vertices.) For a set of colors C with $|C| = k$, a *k-coloring* of G is simply a map $c : V \to C$. In such a coloring, we say that an edge $f \in E$ is *monochromatic* if $\forall u, v \in f : c(u) = c(v)$. A *valid* coloring of G is one in which no edge is monochromatic.

Graph colorings have been widely studied in many forms. The goal is often to show that a valid coloring exists with a small number of colors. Any graph on n vertices clearly has a valid n-coloring; however, for any fixed $k \geq 3$, deciding the existence of a valid k-coloring on general graphs is **NP**-complete.

Theorem 3. *Let $G = (V, E)$ be a hypergraph, and let $k = \min_{f \in E} |f|$. If, for each edge f, there are at most $2^{k-1}/e$ edges $h \in E$ such that $h \cap f \neq \phi$, then there exists a valid 2-coloring of G.*

Proof. We will apply the probablistic method and the LLL to show that a valid 2-coloring exists. Our sample space is the set of all $2^{|V|}$ possible 2-colorings, and for each edge $f \in E$, we define E_f to be the event "f is monochromatic". A valid 2-coloring exists iff $\Pr[\bigwedge_f \overline{E_f}] > 0$.

Let $p = 1/2^{k-1}$ and $d = (2^{k-1}/e) - 1$. For each edge f, $\Pr[E_f] \leq p$ by the fact that f contains at least k vertices. Furthermore, since f intersects at most d edges besides itself, E_f is dependent on at most d of the other events. Therefore, since $e \cdot p \cdot (d+1) = 1$, we can apply the LLL to get $\Pr[\bigwedge_f \overline{E_f}] > 0$. □

2.2 Satisfying Boolean Formulas

Deciding the existence of a satisfying assignment for a Boolean CNF formula is another well-studied (and **NP**-complete) problem. The Lovász Local Lemma can be used in certain cases to show that a satisfying assignment exists.

Definition 2. For a set of Boolean variables $x_1, \ldots, x_n$, a *CNF formula* ϕ has the form $\phi = C_1 \wedge \cdots \wedge C_m$. Each *clause* C_j is an OR of some set of *literals*, where each literal is either x_i or $\overline{x_i}$ for some $i \in [n]$. Clauses C_j and C_k are said to *intersect* if $\exists x_i$ such that both clauses contain either x_i or $\overline{x_i}$. A *satisfying assignment* is a setting of the x_is that makes ϕ evaluate to true.

Theorem 4. *Let $\phi = \bigwedge_{i=1}^{m} C_i$ be a CNF formula, and let $k = min_{C_j \in \phi} |C_j|$. If each clause intersects at most $(2^k/e) - 1$ other clauses, then ϕ is satisfiable.*

Proof. The proof is nearly identical to that of Theorem 3. Our sample space is the set of all 2^n assignments, and for each clause C_j, E_j is the event "C_j is not satisfied". Let $p = 1/2^k$ and $d = (2^k/e) - 1$. Again we have $\forall j$, $\Pr[E_j] \leq p$ by the minimum size of C_j and E_j is dependent on at most d other events by the intersection property. Thus, since $e \cdot p \cdot (d + 1) = 1$, we have $\Pr[\bigwedge_j \overline{E_j}] > 0$, and so a satisfying assignment must exist. □

3 Packet Routing

In this section we set up the packet routing problem, another area in which we will be able to apply the LLL (next lecture). The input is an undirected graph $G = (V, E)$ called the network, and a set of n packets which we represent by pairs of vertices $\{(s_i, t_i) \in V \times V \mid i \in [n]\}$ such that packet k wants to travel from s_k to t_k . We assume that there is a global clock to which all nodes and edges are synchronized, and that at each timestep each edge can carry at most one packet in each direction. The output should be a set of possibly overlapping paths $\{s_i \to t_i \mid i \in [n]\}$ (the "selection" problem), and a schedule to move the packets along the paths while satisfying the edge capacity constraints (the "routing" problem). We will assume that we already have the set of paths, and focus on the routing problem.

Definition 3. Given a set of paths and a schedule:

- the *dilation* is the maximum length of any path
- the *congestion* is the maximum over all edges of the number of packets which cross the edge
- the *makespan* is the maximum over all paths of the time it takes to traverse the path
- the *maximum queue size* is the maximum over all edges and timesteps of the number of packets which are waiting to traverse the edge in the same direction at the same time

Our goal will be to create a schedule with a small makespan and maximum queue size. The makespan must be at least the dilation d and at least the congestion c, so $\Omega(c+d)$ is a lower bound.

Consider the greedy algorithm for producing a schedule: at each timestep, move as many packets as possible while obeying the edge capacities, breaking ties randomly. It is easy to see that the makespan is $O(cd)$ and the maximum queue size is $O(c)$. Furthermore, it is not hard to construct an example for which these bounds on the greedy algorithm are tight.

A paper by Leighton, Maggs & Rao [LMR94] shows how to produce a schedule with makespan $O(c + d)$ and maximum queue size $O(1)$, which is optimal up to constant factors. In the next lecture, we will see a simpler schedule, from the same paper, which has makespan $O((c+d) \cdot 2^{\log^* d})$ and maximum queue size $O(\log d \cdot 2^{\log^* d})$.

As a warmup, consider the following algorithm for producing a schedule: delay each packet randomly and independently for $m \in \{0, \ldots, R\}$ timesteps at the beginning of the schedule, and have each packet go to its destination according to the greedy algorithm after its delay is over. Note that if we allowed the edge capacity constraints to be violated, the schedule would have makespan at most $R+d$. Thus, if we define α to be the maximum queue size, we can convert the invalid schedule (in which edge capacity constraints are violated) to a valid schedule of makespan at most $\alpha(R+d)$ as follows: we simulate each step of the invalid schedule using α steps in the valid schedule. Also note that the final schedule has maximum queue-size at most α.

We now want to choose a value for R and a value β such that, with high probability, the maximum queue size is less than β. That is, we want to say for any fixed edge f and fixed timestep t, Pr[# of packets waiting to take f at time $t \geq \beta$] $\leq p$ for some small enough p. How small do we need p to be to say that $\alpha \leq \beta$ with high probability? We know that there are at most $n \cdot d$ edges we could choose for f and at most $R+d$ timesteps we could choose for t. Thus, if we make $p \leq 1/(nd(R+d))^k$ for some constant k, a union bound will give us $\Pr[\alpha \geq \beta] \leq 1/(nd(R+d))^{k-1}$.

Notice that for any fixed packet, edge and timestep, the probability that the packet reaches the edge at the timestep is equal to $1/R$. Also, the number of packets which cross any fixed edge is bounded by c. Thus, for any fixed edge f and fixed timestep t,

$$\Pr[\#\text{ packets waiting to take } f \text{ at time } t \geq \beta] \leq \binom{c}{\beta} \cdot \left(\frac{1}{R}\right)^{\beta} \leq \left(\frac{e \cdot c}{\beta \cdot R}\right)^{\beta}$$

(using Stirling's approximation). Given this, our goal is now to choose R and β so that $\left(\frac{e \cdot c}{\beta \cdot R}\right)^{\beta} \leq \frac{1}{(nd(R+d))^k}$. First, choose $R = ce$. Then, the goal becomes $\left(\frac{1}{\beta}\right)^{\beta} \leq \frac{1}{(nd(ce+d))^k}$. We now use the fact that $\left(\frac{1}{a}\right)^a = \frac{1}{b} \implies a = \Theta\left(\frac{\log b}{\log\log b}\right)$, and choose $\beta = \Theta\left(\frac{\log((nd(ce+d))^k)}{\log\log((nd(ce+d))^k)}\right) = O\left(\frac{\log(ndc)}{\log\log(ndc)}\right)$. Plugging these values back in, we see that the schedule has makespan $O\left((c+d) \cdot \frac{\log(ndc)}{\log\log(ndc)}\right)$ and maximum queue size $O\left(\frac{\log(ndc)}{\log\log(ndc)}\right)$ with probability at least $1 - \frac{1}{\text{poly}(ndc)}$.

Alternatively, we can set $R = c/\log(nd)$. Then, the goal becomes $\left(\frac{e\log(nd)}{\beta}\right)^{\beta} \leq \frac{1}{(nd(ce+d))^k}$, which is satisfied by setting $\beta = \Theta(\log(nd))$. Plugging these values back in, we see that the final schedule has makespan $O(\log(nd)(\frac{c}{\log(nd)} + d) = O(c + d\log(nd))$ with queue size $O(log(nd))$.

References

[AS00] N. Alon and J.H. Spencer. *The probabilistic method*, chapter 5. Wiley-Interscience, second edition, 2000.

[EL75] P. Erdős and L. Lovász. Problems and results on 3-chromatic hypergraphs and some related questions. *in A. Hajnal, R. Rado, and V. T. Sós, eds.. Infinite and Finite Sets (to Paul Erdős on his 60th birthday)*, II:609–627, 1975.

[LMR94] F.T. Leighton, B. Maggs, and S. Rao. Packet routing and job-shop scheduling in O(congestion+dilation) steps. *Combinatorica*, 14(2):167–180, 1994.