6.7720/18.619/15.070 Lecture 4
Beating the Union Bound: The Lovász Local Lemma

Kuikui Liu

February 12, 2025

Acknowledgements & Disclaimers *In the process of writing these notes, we consulted the classic textbook titled "The Probabilistic Method" by Noga Alon and Joel Spencer, as well as materials by Jan Vondrak. Please be advised that these notes have not been subjected to the usual scrutiny reserved for formal publications. If you do spot an error, please contact the instructor.*

1 Boolean Satisfiability

In this lecture, we use the probabilistic method to study perhaps the quintessential algorithmic problem in computer science: SAT. Our goal will also be to introduce a famous technique, known as the *Lovász Local Lemma (LLL)*, for lower bounding the probability that a collection of "bad events" are simultaneously avoided. This technique is often employed in conjunction with the probabilistic method. More generally, it gives a way of "beating the union bound" in settings where there is "bounded dependence" between events (but not full independence).

To define the SAT problem, we must first define what a CNF-formula is. Let $\mathcal{V}$ be a collection of Boolean-valued *variables*, i.e. variables taking values in $\{\mathsf{T}, \mathsf{F}\}$, where T denotes "True" and F denotes "False".

- For a variable $x \in \mathcal{V}$, its *negation* is the expression $\neg x$, which on input T (resp. F) *evaluates* to F (resp. T). A *literal* is an expression of the form x or $\neg x$ for some variable $x \in \mathcal{V}$.
- A *clause* C is an expression given by an OR of a collection of literals. For example, we could have $C = x_1 \vee \neg x_2$, which evaluates to F on inputs $(x_1 \leftarrow \mathsf{F}, x_2 \leftarrow \mathsf{T})$ and evalutes to T on all other inputs.
- A CNF-*formula*[1] $\Phi = (\mathcal{V}, \mathcal{C})$ is an expression over the variables in $\mathcal{V}$ given by an AND of a collection of clauses $\mathcal{C}$. For example, we could have

  $$(x_1 \vee \neg x_2) \wedge (x_2 \vee \neg x_3) \wedge (x_3 \vee \neg x_1),$$

  where there are three variables $\mathcal{V} = \{x_1, x_2, x_3\}$ and three clauses $x_1 \vee \neg x_2, x_2 \vee \neg x_3, x_3 \vee \neg x_1$. This particular formula evaluates to T on the assignments $(x_1 \leftarrow \mathsf{T}, x_2 \leftarrow \mathsf{T}, x_3 \leftarrow \mathsf{T})$ and $(x_1 \leftarrow \mathsf{F}, x_2 \leftarrow \mathsf{F}, x_3 \leftarrow \mathsf{F})$; it evaluates to F on all other assignments.
- We say an assignment of T/F to the variables *satisfies* a clause C if C evaluates to T under the assignment; note that this holds if and only if at least one literal in C evaluates to T. Similarly, we say an assignment satisfies a CNF-formula $\Phi = (\mathcal{V}, \mathcal{C})$ if it satisfies every clause $C \in \mathcal{C}$. We say the CNF-formula Φ is *satisfiable* if there exists at least satisfying assignment for Φ.

The SAT problem is the problem of finding a satisfying assignment given a CNF-formula. This is one of the most fundamental algorithmic problems in computer science, and is among the most basic and well-studied of all *constraint satisfaction problems* in discrete mathematics. It is the core problem underpinning the theory of NP-*completeness*, which allows us to rigorously classify a wide variety of computational problems as either "efficiently solvable" (i.e. "easy") or "computationally intractable" (i.e. "hard").

[1] Here, CNF stands for *conjunctive normal form.*

It is believed that SAT is a hard problem to solve in the worst case; in fact, simply deciding whether or not a CNF-formula is satisfiable at all is hard.[2] Perhaps surprisingly, even if we restrict the CNF-formula to be k*-uniform*, i.e. by requiring that all clauses have exactly k distinct variables, the problem remains just as hard as the full SAT problem as long as $k \geq 3$. Hence, it makes sense to ask the following two questions.

- Given a CNF-formula with n variables and m clauses, let OPT denote the maximum fraction of satisfied clauses over all possible assignments. For fixed $\alpha > 0$, can we efficiently find an assignment satisfying at least an $\alpha \cdot \mathsf{OPT}$ fraction of clauses?
- Can we enforce additional structure on our CNF-formulas so that we are guaranteed they are satisfiable and we can efficiently find a satisfying assignment?

Regarding the first question, let us consider the version of SAT in which our CNF-formulas are restricted to be 3-uniform. Despite the fact that this restricted problem remains as hard as general SAT, we can always find assignments which satisfy a large fraction of clauses.

Proposition 1.1. *Given any* 3*-uniform* CNF*-formula* $\Phi = (\mathcal{V}, \mathcal{C})$ *with* n *variables and* m *clauses, there exists an assignment satisfying at least* $\frac{7}{8} \cdot m$ *clauses irrespective of whether or not* Φ *is satisfiable.*

Remark 1. Furthermore, there is an efficient deterministic polynomial-time algorithm for finding such an assignment based on the *method of conditional expectations*. If one slightly relaxes the 3-uniformity constraint to allow for clauses with possibly one or two literals, then there is still an efficient algorithm due to Karloff–Zwick for finding an assignment satisfying at least $\frac{7}{8} \cdot m$ clauses, at least in the case where Φ is satisfiable [KZ97]. However, the algorithm is much more involved.

Remark 2. Amazingly, it is known that a *computational phase transition* occurs in the sense that for any arbitrarily small positive constant $\epsilon > 0$, the problem of finding an assignment satisfying at least $\left(\frac{7}{8} + \epsilon\right) \cdot m$ clauses remains just as hard as finding perfectly satisfying solutions to general CNF-formulas. This is one of the seminal hardness-of-approximation result due to Håstad [Hås01], who built upon the famous PCP Theorem, a landmark result in theoretical computer science [Fei+96; AS98; Aro+98].

Proof. We prove existence by picking a uniformly random assignment and showing that it has a nonzero probability of satisfying at least $\frac{7}{8} \cdot m$ clauses. In fact, we will show that it satisfies $\frac{7}{8} \cdot m$ clauses in expectation, which is a stronger claim. For each clause $C \in \mathcal{C}$, let $\mathbf{1}_C$ denote the indicator random variable for whether or not the clause C is satisfied by a given assignment. By linearity of expectation, the expected number of satisfied clauses is

$$\sum_{C \in \mathcal{C}} \mathbb{E}[\mathbf{1}_C] = \sum_{C \in \mathcal{C}} \Pr[C \text{ is satisfied}],$$

where the expectations and probabilities are with respect to a uniformly random assignment $x \sim \mathsf{Unif}\{\mathsf{T}, \mathsf{F}\}^{\mathcal{V}}$. Since each clause C is an OR of exactly 3 literals, exactly one out of the eight possible assignments to its 3 constituent variables fails to satisfy C. In particular, $\Pr[C \text{ is satisfied}] = 7/8$ for every $C \in \mathcal{C}$ and we are done. □

2 SAT and the Lovász Local Lemma (LLL)

Let us now turn to finding natural and generic conditions under which a CNF-formula is guaranteed to be satisfiable.

Theorem 2.1. *Fix* $k \in \mathbb{N}$*, and let* $\Phi = (\mathcal{V}, \mathcal{C})$ *be a* k*-uniform* CNF*-formula with* n *variables and* m *clauses. If every variable occurs in at most* $\frac{2^k}{4k}$ *clauses, then* Φ *is satisfiable.*

Remark 3. Such a satisfying assignment can also be efficiently found in polynomial-time with high probability using a stochastic local search algorithm due to Moser–Tardos [MT10]. In the literature, this is often referred to as the *algorithmic Lovász Local Lemma.*

[2] However, most instances of SAT one encounters in practice, even those with millions of constraints, are easily dispatched by modern SAT-solving software.

2.1 Beating the Union Bound with Bounded Dependence

Suppose we have a large collection of events $A_1, \dots, A_m$, and we wish to certify that with positive probability, none of the events occur (i.e. $\Pr\left[\bigcap_{i=1}^m \overline{A}_i\right] > 0$, where we write $\overline{A}$ for the complementary event). We typically view $A_1, \dots, A_m$ as "bad events" we want to avoid, e.g. if we pick a random assignment from $\{\mathsf{T}, \mathsf{F}\}^n$, then we could take A_i to be the event that the ith clause is not satisfied. If the events were all jointly independent, then of course $\Pr\left[\bigcap_{i=1}^m \overline{A}_i\right] = \prod_{i=1}^m (1 - \Pr[A_i])$, which is positive if and only if none of the events individually happen with probability 1.

However, in most applications, the events $A_1, \dots, A_m$ are not jointly independent. One could use the Union Bound to say that $\Pr\left[\bigcap_{i=1}^m \overline{A}_i\right] = 1 - \Pr\left[\bigcup_{i=1}^m A_i\right] \geq 1 - \sum_{i=1}^m \Pr[A_i]$, but this requires $\sum_{i=1}^m \Pr[A_i] < 1$ which is often far too restrictive. For instance, if we take A_i to be the event that a uniformly random assignment satisfies the ith clause in a k-uniform CNF-formula, then

$$\Pr[A_i] = 2^{-k}, \qquad \forall i = 1, \dots, m.$$

This is a constant failure probability for constant k, which is much larger than the $1/m$ probability needed in order for the Union Bound to be effective.

The famous *Lovász Local Lemma (LLL)* allows us to significantly "beat the Union Bound" assuming the collection of events $\{A_i\}_{i=1}^m$ are "not too dependent" on each other. It is a fundamental tool in discrete probability and theoretical computer science, and often goes hand-in-hand with the *probabilistic method*.

Definition 1 (Mutual Independence). *Let $A_1, \dots, A_m$ be a collection of events within the same probability space. For each $i \in [m]$ and $J \subseteq [m] \setminus \{i\}$, we say A_i is* mutually independent *of $\{A_j : j \in J\}$ if*

$$\Pr\left[A_i \cap \bigcap_{j \in J'} A_j\right] = \Pr[A_i] \cdot \Pr\left[\bigcap_{j \in J'} A_j\right], \qquad \forall J' \subseteq J.$$

Definition 2 (Dependency Graph). *Let $A_1, \dots, A_m$ be a collection of events within the same probability space. We say an (undirected) graph $G = (V, E)$ with $V \cong [m]$ is a* dependency graph *for the events $\{A_i\}_{i=1}^m$ if for every $v \in V$, the event A_v is mutually independent of the events $\{A_u : u \notin N[v]\}$, where $N[v] \stackrel{\mathsf{def}}{=} \{v\} \cup \{u \in V : uv \in E\}$ denotes the* closed neighborhood of v.

Remark 4. We emphasize that a collection of events $\{A_i\}_{i=1}^m$ need not have a *unique* dependency graph. For instance, the complete graph K_m is always a valid dependency graph w.r.t. any collection of events $\{A_i\}_{i=1}^m$, although this is not very useful since its vertices have large degree. One can also consider directed dependency graphs, but we will not do so here. For further discussion of these points, see e.g. [SS05].

Theorem 2.2 ((Symmetric) Lovász Local Lemma (LLL)). *If $\{A_i\}_{i=1}^m$ admit a dependency graph of maximum degree $\leq d$, and $\Pr[A_i] \leq \frac{1}{e(d+1)}$ for all $i = 1, \dots, m$, then $\Pr\left[\bigcap_{i=1}^m \overline{A}_i\right] > 0$.*

Remark 5. As we will see from the proof, we can achieve a lower bound of $\left(1 - \frac{1}{d+1}\right)^n$. There are stronger versions of the LLL which make weaker assumptions and achieve more precise quantitative lower bounds. There are beautiful and deep connections between these refined versions of the LLL, statistical physics, and *zero-freeness* of the so-called *multivariate independence polynomial* attached to the dependency graph; see e.g. [SS05]. The version stated in Theorem 2.2 is among the simplest and most "user-friendly".

Before we prove the Lovász Local Lemma, let us first apply it to satisfiability of k-uniform CNF-formulas.

Proof of Theorem 2.1. For an arbitrary k-uniform CNF-formula $\Phi = (\mathcal{V}, \mathcal{C})$, and let $\boldsymbol{x} \sim \mathsf{Unif}\{\mathsf{T}, \mathsf{F}\}^{\mathcal{V}}$ be a uniformly random assignment. If $C_1, \dots, C_m$ is some ordering of the clauses of $\mathcal{C}$, then let A_i denote the event that clause C_i is not satisfied by $\boldsymbol{x}$, for each $i = 1, \dots, m$. Our goal is to show that $\Pr\left[\bigcap_{i=1}^m \overline{A}_i\right] > 0$, i.e. that there is some positive probability that $\boldsymbol{x}$ satisfies all clauses simultaneously. This is enough to certify Φ is satisfiable.

To apply Theorem 2.2, we verify the two conditions of the lemma. The key is to observe that for any clause C_i, if we let

$$J = \{j \in [m] : C_j \text{ does not share any variables with } C_i\},$$

then A_i is mutually independent of $\{A_j\}_{j \in J}$. Hence, if we consider the graph G whose vertices correspond to the events $A_1, \ldots, A_m$, and where we connected $A_i \sim A_j$ by an edge if and only if C_i, C_j share at least one variable, then G is a valid dependency graph for $A_1, \ldots, A_m$. To bound its maximum degree, observe that by k-uniformity of Φ plus our assumption that every variable is contained in at most $\frac{2^k}{4k}$ clauses,

$$|[m] \setminus J| \leq \frac{2^k}{4k} \cdot k = \frac{1}{4} \cdot 2^k \stackrel{\mathsf{def}}{=} d.$$

Since C_i was arbitrary, the maximum degree of the aforementioned dependency graph is at most $\frac{1}{4} \cdot 2^k$. On the other hand, since each clause has exact k literals and $\boldsymbol{x}_v \sim \mathsf{Unif}\{\mathsf{T}, \mathsf{F}\}$ independently for each variable $v \in \mathcal{V}$,

$$\Pr[A_i] \leq 2^{-k} \leq \frac{1}{e\,(d+1)}.$$

Invoking Theorem 2.2 then completes the proof.[3] □

2.2 Proof of Theorem 2.2

Expanding using conditional probabilities, observe that

$$\Pr\left[\bigcap_{i=1}^{n} \overline{A}_i\right] = \prod_{i=1}^{n} \Pr\left[\overline{A}_i \,\middle|\, \bigcap_{j=1}^{i-1} \overline{A}_j\right] = \prod_{i=1}^{n} \left(1 - \Pr\left[A_i \,\middle|\, \bigcap_{j=1}^{i-1} \overline{A}_j\right]\right).$$

We are done if we can show that $\Pr\left[A_i \,\middle|\, \bigcap_{j=1}^{i-1} \overline{A}_j\right] < 1$ for all i. We've already assumed that without any conditioning, $\Pr[A_i] \leq \frac{1}{e(d+1)}$, and so the main task is to show that conditioning does not degrade this bound too much when A_i is dependent on at most d other events. This is formalized in the following lemma.

Lemma 2.3. *Suppose $A_1, \ldots, A_m$ satisfy the assumptions of Theorem 2.2. Then for any $i = 1, \ldots, m$ and $J \subseteq [m] \setminus \{i\}$,*

$$\Pr\left[A_i \,\middle|\, \bigcap_{j \in J} \overline{A}_j\right] \leq \frac{1}{d+1}.$$

Proof. We go by increasing induction on the size of J. The case $|J| = 0$ is immediate by assumption. Suppose the claim holds for any J of cardinality $|J| \leq k$, for some $k \geq 0$. For the inductive step, let $J \subseteq [m] \setminus \{i\}$ have size $k+1$. Let $N(i)$ denote the neighbors of i in the dependency graph, i.e. the events on which A_i is dependent. Let us divide J into two sets $J_{\mathsf{dep}} = J \cap N(i)$, $J_{\mathsf{indep}} = J \setminus N(i)$. Note that we may assume J_{dep} is nonempty, since otherwise, the conditional probability of interest is automatically upper bounded by $\frac{1}{e(d+1)}$ by assumption. By Bayes' Rule,

$$\Pr\left[A_i \,\middle|\, \bigcap_{j \in J} \overline{A}_j\right] = \frac{\Pr\left[A_i \cap \bigcap_{j \in J_{\mathsf{dep}}} \overline{A}_j \,\middle|\, \bigcap_{j \in J_{\mathsf{indep}}} \overline{A}_j\right]}{\Pr\left[\bigcap_{j \in J_{\mathsf{dep}}} \overline{A}_j \,\middle|\, \bigcap_{j \in J_{\mathsf{indep}}} \overline{A}_j\right]}.$$

[3]Technically, when $d \leq 2$ (i.e. $k \leq 3$), one should instead use the version of the LLL which requires $\Pr[A_i] \leq \frac{1}{4d}$ for all $i = 1, \ldots, m$.

For the numerator, we have

$$\begin{aligned}\Pr\left[A_i \cap \bigcap_{j \in J_{\mathsf{dep}}} \overline{A}_j \,\middle|\, \bigcap_{j \in J_{\mathsf{indep}}} \overline{A}_j\right] &\le \Pr\left[A_i \,\middle|\, \bigcap_{j \in J_{\mathsf{indep}}} \overline{A}_j\right] \\ &= \Pr[A_i] && (A_i \text{ is independent of } \{A_j\}_{j \in J_{\mathsf{indep}}}) \\ &\le \frac{1}{e(d+1)}.\end{aligned}$$

For the denominator, if we order J_{dep} as $j_1, \dots, j_t$ for $t \le d$, then

$$\begin{aligned}\Pr\left[\bigcap_{j \in J_{\mathsf{dep}}} \overline{A}_j \,\middle|\, \bigcap_{j \in J_{\mathsf{indep}}} \overline{A}_j\right] &= \prod_{s=1}^{t} \Pr\left[\overline{A}_{j_s} \,\middle|\, \bigcap_{j \in J_{\mathsf{indep}}} \overline{A}_j \cap \bigcap_{p=1}^{s-1} \overline{A}_{j_p}\right] \\ &= \prod_{s=1}^{t} \left(1 - \Pr\left[A_{j_s} \,\middle|\, \bigcap_{j \in J_{\mathsf{indep}}} \overline{A}_j \cap \bigcap_{p=1}^{s-1} \overline{A}_{j_p}\right]\right) \\ &\ge \left(1 - \frac{1}{d+1}\right)^t && \text{(Induction Hypothesis)} \\ &\ge \left(1 - \frac{1}{d+1}\right)^d && \text{(Using the maximum degree bound)} \\ &\ge \frac{1}{e}.\end{aligned}$$

Combining the preceding two displays completes the induction step, and hence, the proof. □

References

[Aro+98] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. "Proof verification and the hardness of approximation problems". In: *J. ACM* 45.3 (May 1998), pp. 501–555. ISSN: 0004-5411. DOI: 10.1145/278298.278306 (cit. on p. 2).

[AS98] Sanjeev Arora and Shmuel Safra. "Probabilistic checking of proofs: a new characterization of NP". In: *J. ACM* 45.1 (Jan. 1998), pp. 70–122. ISSN: 0004-5411. DOI: 10.1145/273865.273901 (cit. on p. 2).

[Fei+96] Uriel Feige, Shafi Goldwasser, Laszlo Lovász, Shmuel Safra, and Mario Szegedy. "Interactive proofs and the hardness of approximating cliques". In: *J. ACM* 43.2 (Mar. 1996), pp. 268–292. ISSN: 0004-5411. DOI: 10.1145/226643.226652 (cit. on p. 2).

[Hås01] Johan Håstad. "Some Optimal Inapproximability Results". In: *J. ACM* 48.4 (July 2001), pp. 798–859. ISSN: 0004-5411 (cit. on p. 2).

[KZ97] H. Karloff and U. Zwick. "A 7/8-approximation algorithm for MAX 3SAT?" In: *Proceedings 38th Annual Symposium on Foundations of Computer Science*. 1997, pp. 406–415. DOI: 10.1109/SFCS.1997.646129 (cit. on p. 2).

[MT10] Robin A. Moser and Gábor Tardos. "A Constructive Proof of the General Lovász Local Lemma". In: *J. ACM* 57.2 (Feb. 2010). ISSN: 0004-5411. DOI: 10.1145/1667053.1667060 (cit. on p. 2).

[SS05] Alexander D. Scott and Alan D. Sokal. "The Repulsive Lattice Gas, the Independent-Set Polynomial, and the Lovász Local Lemma". In: *Journal of Statistical Physics* 118.5 (2005), pp. 1151–1261 (cit. on p. 3).