CS 880: Pseudorandomness and Derandomization 2/18/2013

Lecture 8: Applications of Small-Bias Generators

Instructors: Holger Dell and Dieter van Melkebeek Scribe: Gautam Prakriya

In the previous lecture, we developed the basics of harmonic analysis and used them to quantitatively show that every small-bias generator is almost k-wise uniform. This led to an ϵ-PRG for k-wise uniformity with seed length $O(k + \log(1/\epsilon) + \log r)$, where r denotes the output length. In this lecture we give a construction that improves the dependence on r to $O(\log \log r)$, and for this we again use small-bias generators. In general, it is a good idea to consider small-bias generators whenever harmonic analysis is involved in the correctness of the construction. This is because of the close connection between the Fourier coefficients of a distribution and its bias that we mentioned in the last lecture.

We also present two other applications of small-bias generators, namely in the context of approximation algorithms: a hardness result and a (derandomized) algorithm for the maximum satisfiability problem of a set of quadratic equations over $\mathbb{F}_2$.

1 Almost k-Wise Uniform Generators

In this section, we develop an alternate way to construct an almost k-wise uniform generator. Theorem 5 from Lecture 4 gives a construction of a perfect k-wise uniform generator $G : \{0,1\}^\ell \to \{0,1\}^r$ with seed length $\ell = O(k \cdot \log r)$. We can try to further reduce the need for truly random bits by derandomizing G, i.e., by using another PRG G' to generate the seed of length ℓ for G.

What second PRG G' should we use? The answer depends on (i) what properties of the final pseudorandom distribution D matter, and (ii) how G uses its seed. As for (i), by the analysis of last lecture, we know that it is sufficient for D to have small bias in order to get a distribution that is almost k-wise uniform. As for (ii), the PRG G in Theorem 5 of Lecture 4 (which is based on the one in Lemma 4 of that lecture) is a linear mapping, and in that case the following lemma shows a simple connection between the biases of $D = G(D')$ and the biases of D'. Facts (i) and (ii) combined then suggest the use of a small-bias generator G' to generate D'.

Proposition 1. *Let D' be a distribution on $\{0,1\}^\ell$ and let $D \doteq M \cdot D'$, where $M \in \{0,1\}^{r \times \ell}$. For every $a \in \{0,1\}^r$, we have*

$$\mathrm{bias}_a(D) = \mathrm{bias}_{M^T \cdot a}(D') \,.$$

Proof. We have

$$\mathrm{bias}_a(D) = \mathbb{E}\left[(-1)^{\langle a, D\rangle}\right] = \mathbb{E}\left[(-1)^{\langle a, M \cdot D'\rangle}\right] = \mathbb{E}\left[(-1)^{\langle M^T \cdot a, D'\rangle}\right] = \mathrm{bias}_{M^T \cdot a}(D') \,,$$

where the third equality follows because

$$\langle a, MD'\rangle = a^T(MD') = (a^T M)D' = (M^T a)^T D' = \langle M^T a, D'\rangle. \qquad \square$$

Proposition 1 shows that, if D' has small bias, then all the biases of $D = M \cdot D'$ are equally small, except with respect to those a for which $M^T a = 0$ (for which it is 1). In case the linear mapping M

defines a k-wise uniform generator, we argued in Lecture 4 (cf. the proof of Proposition 3) that every k rows of M are linearly independent, or equivalently, every k columns of M^T are linearly independent. Thus, the only vector $a \in \{0,1\}^r$ with at most k nonzero entries that satisfies $M^T a = 0$ is the zero vector. In combination with Proposition 2 from Lecture 7, this means that D is close to being k-wise uniform in the following quantitative sense.

Proposition 2. *Let D' be a β-bias distribution on $\{0,1\}^\ell$, and let $G : \{0,1\}^\ell \to \{0,1\}^r$ be a linear k-wise uniform generator. Then $D \doteq G(D')$ has the following properties for every $I \subseteq [r]$ with $|I| = k$:*

(i) $(\forall y \in \{0,1\}^k) \quad \Big|\Pr[D|_I = y] - \Pr[U_k = y]\Big| \leq \big(1 - \frac{1}{2^k}\big) \cdot \beta$, *and*

(ii) $d_{\text{stat}}(D|_I, U_k) \leq \sqrt{2^k - 1} \cdot \frac{\beta}{2}$.

Recall that Theorem 4 from Lecture 6 yields a generator G' with seed length $O(\log(\ell/\beta))$ that produces a β-bias distribution D' on $\{0,1\}^\ell$. The construction of Theorem 5 from Lecture 4 gives us a linear k-wise uniform generator on $\{0,1\}^r$ with seed length $\ell = O(k \cdot \log r)$. Combining those two constructions as in Proposition 2 with $\beta = \epsilon/\sqrt{2^k - 1}$ yields the following result.

Theorem 3. *For all integers r and $k \in [r]$, and every real $\epsilon > 0$, there exists an efficiently computable ϵ-pseudorandom generator $G : \{0,1\}^\ell \to \{0,1\}^r$ for k-wise uniformity with $\ell = O(k + \log(1/\epsilon) + \log\log r)$.*

In many settings where k-wise uniformity is sufficient, almost k-wise uniformity is also sufficient. This allows us to reduce the seed length from $O(k \cdot \log r)$ down to $O(k + \log(1/\epsilon) + \log\log r)$, where ϵ denotes the allowed statistical distance from k-wise uniform. Note that this improves over the seed length $O(k + \log(1/\epsilon) + \log r)$ from last lecture. For an example of an application where the reduction from $\log(r)$ down to $\log\log(r)$ matters, we refer to [MNS95, Section 8].

Finally, let us point out that, in some settings, having *perfect* rather than *almost* k-wise uniformity is critical. An example is the additivity of the variance given by Proposition 3 of Lecture 5 and the applications that depend on it. The variance of the sum of t samples of a random variable grows only linearly with t when the samples are pairwise independent, but quadratically when the samples are merely almost pairwise independent. This is because the $O(t^2)$ cross terms in the proof of Proposition 3 no longer vanish.

2 Hardness of Approximation

In this section we study the problem of finding approximate solutions to a system of quadratic equations over $\mathbb{F}_2$ in the following sense: Given a list S of m polynomials Q_i of degree at most two in n variables over $\mathbb{F}_2$, our goal is to find an assignment $x \in \{0,1\}^n$ that satisfies many of the equations $Q_i(x) = 0$. The best we could hope for is to find a solution that satisfies a fraction

$$\text{Max-QE}(S) \doteq \frac{1}{m} \max_{x \in \{0,1\}^n} \sum_{i=1}^{m} I[Q_i(x) = 0]$$

of the equations. Finding a solution that realizes Max-QE(S) is NP-hard. In fact, it is even NP-hard to decide whether Max-QE(S) $= 1$ holds.

Proposition 4. *Given a system S of quadratic equations over $\mathbb{F}_2$, deciding whether there exists an assignment to the variables that satisfies all the equations is* NP*-hard.*

Proof. To show that the problem is NP-hard, we reduce 3-SAT to it based on the following observation. Every clause in a 3-CNF formula φ can represented by a polynomial of degree 3 over $\mathbb{F}_2$, which in turn can be represented by a pair of polynomials of degree 2 with the introduction of a new variable. For instance, consider the clause $(x_1 \vee \overline{x_2} \vee x_3)$. An assignment satisfies this clause if and only if it satisfies $(1-x_1)\cdot x_2 \cdot (1-x_3) = 0$. This equation is equivalent to the system of two quadratic equations

$$\begin{cases} (x_2 - y_{1,2})(1-x_3) = 0 \\ y_{1,2} - x_1 x_2 = 0, \end{cases}$$

where $y_{1,2}$ is an auxiliary variable. By collecting these equations for all clauses of φ, we obtain a system S of quadratic equations that is satisfiable if and only if φ is satisfiable. The transformation from φ into S only takes polynomial time. □

Given Proposition 4, we relax our goal to finding an assignment that satisfies at least a fraction $\alpha \cdot \text{Max-QE}(S)$ of the equations for some constant $\alpha > 0$. We will use small-bias generators to investigate (1) for which α the relaxed goal remains NP-hard, and (2) for which α we can realize the goal in polynomial time. Results of type (1) are called inapproximability results, and results of type (2) α-approximation algorithms.

2.1 Inapproximability

We show that finding an assignment that satisfies a fraction $\alpha \cdot \text{Max-QE}(S)$ of the equations is NP-hard for every constant $\alpha > \frac{1}{2}$. In fact, we prove a somewhat stronger statement.

Theorem 5. *For every $\alpha \doteq \frac{1}{2} + \epsilon$ where ϵ is a positive real, it is* NP*-hard to distinguish between the cases* $\text{Max-QE}(S') = 1$ *and* $\text{Max-QE}(S') < \alpha$*, where S' is the input, a list of polynomials of degree at most two over $\mathbb{F}_2$.*

Proof. Given Proposition 4, it suffices to construct a polynomial-time transformation f that takes a system S of quadratic equations and produces a system $S' = f(S)$ of quadratic equations with the following properties:

- Completeness: $\text{Max-QE}(S) = 1$ implies $\text{Max-QE}(S') = 1$.
- Soundness: $\text{Max-QE}(S) < 1$ implies $\text{Max-QE}(S') < \alpha$.

Here is how we construct S'. Let the system S consist of the equations $Q_i = 0$ for $i \in [m]$, where Q_i denotes a polynomial of degree at most 2 in n variables over $\mathbb{F}_2$. Fix an assignment $x \in \{0,1\}^n$, and define

$$I_i \doteq I[Q_i(x) \neq 0]$$

to be the indicator that the ith equation is not satisfied. Note that x satisfies all the equations of S if and only if $I \doteq (I_1, \cdots, I_m) = 0$.

Consider the linear combination $\sum c_i Q_i$ for some $c \in \{0,1\}^m$. Since we are working over $\mathbb{F}_2$, a sum of equations is satisfied by x if and only if the number of equations that are not satisfied by x is even. In terms of the above linear combination, we have that

$$\sum_{i=1}^{m} c_i Q_i(x) = 0 \iff \sum_{i=1}^{m} c_i I_i = 0 \iff \langle c, I \rangle = 0,$$

where all the arithmetic is over $\mathbb{F}_2$. Therefore, for a uniform linear combination we have

$$\Pr_{c \leftarrow U_m}[\sum_{i=1}^{m} c_i Q_i(x) = 0] = \Pr_{c \leftarrow U_m}[\langle c, I \rangle = 0] = \begin{cases} 1 & \text{if } I = 0 \\ \frac{1}{2} & \text{otherwise.} \end{cases}$$

If we defined S' to consist of all equations of the form $\sum_{i=1}^{m} c_i Q_i(x) = 0$ for all possible $c \in \{0,1\}^m$, we would be close to our goal; the only issue is that S' is too large to be constructed in polynomial time. To make the size of S' polynomial in m, we let c range over the image of a small-bias generator, i.e., over $G(U_\ell)$ where $G : \{0,1\}^\ell \to \{0,1\}^m$ is a β-bias generator. By the definition of bias and the connection with PRGs for polynomials of degree 1, for $D_m \doteq G(U_\ell)$, we have that

$$\Pr_{c \leftarrow D_m}[\sum_{i=1}^{m} c_i Q_i(x) = 0] = \Pr_{c \leftarrow D_m}[\langle c, I \rangle = 0] \begin{cases} = 1 & \text{if } I = 0 \\ \leq \frac{1}{2} + \frac{\beta}{2} & \text{otherwise.} \end{cases}$$

If we set $\beta < 2\epsilon$, we meet the completeness and soundness conditions of the transformation. Moreover, if we use the small-bias generator from Theorem 4 of Lecture 6, the number of equations in S' is $2^\ell = O((\frac{m}{\epsilon})^2)$, which is is polynomial, and we can construct each individual equation in polynomial time. Thus, we have realized the transformation f we needed. This completes the proof. □

Theorem 5 immediately yields the following corollary about the hardness of approximating the value Max-QE(S).

Corollary 6. *For every $\alpha \doteq \frac{1}{2} + \epsilon$ where ϵ is a positive real, the following problem is* NP*-hard: Given a list S' of polynomials of degree at most 2 over $\mathbb{F}_2$, find a value v such that*

$$\alpha \cdot \text{Max-QE}(S') \leq v \leq \text{Max-QE}(S').$$

Proof. Given an approximation algorithm as in the statement of the corollary, we can distinguish between the two cases of systems S' in Theorem 5 as follows: Run the approximation algorithm on S' and check whether the value v it outputs is at least α. If so, we are in the first case, and otherwise in the second case. □

Corollary 6 implies that it is NP-hard to find an assignment for the variables of a given system S of quadratic equations that satisfies a fraction at least $\alpha \cdot \text{Max-QE}(S)$ of the equations for any α of the form $\alpha = \frac{1}{2} + \epsilon$ where ϵ is a positive constant.

In fact, the proofs show that we can pick ϵ to be a decreasing function of the input size – as long as $1/\epsilon$ is polynomially bounded, the proofs of Theorem 5 and Corollary 5 carry through.

One can do substantially better using a considerably more involved approach based on harmonic analysis. That approach shows that α in Theorem 5 can be chosen as $\alpha = \frac{3}{8} + \epsilon$, where $1/\epsilon$ is polynomially bounded [Hås11]. The same improvement to Corollary 6 follows.

Corollary 6 can be further improved by considering as the first case in Theorem 5 systems S that are highly satisfiable but not necessarily fully satisfiable. The approach based on harmonic analysis then shows that α in Corollary 6 can be chosen as $\alpha = \frac{1}{4} + \epsilon$, where $1/\epsilon$ is polynomially bounded [Hås01]. The latter result is tight in the sense that $\alpha = \frac{1}{4}$ can be realized in polynomial time, as we will see next.

2.2 Approximation algorithms

In order to construct a good approximation algorithm, we first analyze how well a random assignment fares, and then show how to deterministically find an assignment that does at least as well.

Let us first focus on a single equation $Q = 0$, where Q is a polynomial of degree at most 2 in n variables over $\mathbb{F}_2$. What is the minimum probability that a uniform assignment $x \in_u \{0,1\}^n$ satisfies $Q(x) = 0$?

Well, if $Q \equiv 1$, then the probability is zero. Such unsatisfiable equations can be recognized in deterministic polynomial time, even when they are given in the form of an arithmetic formula or circuit. Recognizing such Q is equivalent to the polynomial identity testing problem from Lecture 1. In general, the only efficient algorithms for this problem are randomized. However, for degree two there is a simple randomized algorithm: Evaluate the formula or circuit symbolically, keeping track of the list of all monomials, and check whether at the end only the constant monomial is present. Since there are only $O(n^2)$ monomials of degree at most 2, this can be done in deterministic polynomial time.

How low can the probability be for a satisfiable equation? For $Q(x) = x_1x_2 - 1$, the probability that a uniform assignment satisfies $Q = 0$ equals $\frac{1}{4}$. We leave it as an exercise to argue that this is as low as that probability gets.

Exercise 1. *Prove that a uniform assignment to the variables of a satisfiable quadratic equation over $\mathbb{F}_2$ is satisfying with probability at least $\frac{1}{4}$.*

Suppose S contains m equations that are individually satisfiable. By linearity of expectation, the expected number of equations that a uniform assignment satisfies is at least $\frac{m}{4}$. This implies the existence of an assignment x that satisfies at least that many equations. Since no assignment can satisfy any of the other equations, that assignment satisfies at least a fraction $\frac{1}{4}$Max-QE(S) of the equations. What remains to show is to find such an assignment x in deterministic polynomial time.

Suppose that we generate the assignment x using an ϵ-PRG G for polynomials of degree 2. For each of the m individually satisfiable equations $Q = 0$, the probability that x satisfies it is at least $\frac{1}{4} - \epsilon$. Thus, the expected number of satisfied equations among those m is at least $(\frac{1}{4} - \epsilon)m$, and therefore there exists at least one x in the image of G that satisfies $\lceil(\frac{1}{4} - \epsilon)m\rceil$ of the equations. Now, if ϵ is sufficiently small, the latter quantity is at least $\frac{m}{4}$. We leave it as an exercise to show that $\epsilon < \frac{1}{4m}$ suffices to guarantee that property.

Exercise 2. *Prove that if $\epsilon < \frac{1}{4m}$ then $\lceil(\frac{1}{4} - \epsilon)m\rceil \geq \frac{m}{4}$.*

Corollary 4 from Lecture 6 gives us an efficiently computable ϵ-PRG G for degree 2 polynomials with seed length $\ell = O(\log(m/\epsilon))$. For $\epsilon = \frac{1}{5m}$, we can cycle over all assignments x in the range of G in time polynomial in m. Trying out all those assignments and selecting the one that satisfies the most equations yields the following result.

Theorem 7. *There exists a deterministic polynomial-time algorithm that takes a list S of polynomials Q of degree at most 2 over $\mathbb{F}_2$ and outputs an assignment that satisfies at least $\frac{1}{4} \cdot \text{Max-QE}(S)$ of the equations $Q = 0$.*

Theorem 7 shows the optimality of the strengthening of Corollary 6 we mentioned, in which $\frac{1}{2}$ is replaced by $\frac{1}{4}$.

In the case where the system S is fully satisfiable, it turns out that one can efficiently reduce to a fully satisfiable system S' in which every equation is individually satisfiable with probability at least $\frac{3}{8}$ [Hås11]. The same approach as above then yields a deterministic polynomial-time algorithm to find an assignment that satisfies a fraction at least $\frac{3}{8}$ of the equations. This shows the optimality of the strengthening of Theorem 5 we mentioned, in which $\frac{1}{2}$ is replaced by $\frac{3}{8}$.

References

[Hås01] Johan Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001.

[Hås11] Johan Håstad. Satisfying degree-d equations over GF[2]^n. In *Proceedings of the 14th International Workshop on Approximation, Randomization, and Combinatorial Optimization, APPROX-RANDOM 2001*, pages 242–253, 2011.

[MNS95] Alain Mayer, Moni Naor, and Larry Stockmeryer. Local computations on static and dynamic graphs. In *Proceedings of the Third Israel Symposium on the Theory of Computing and Systems*, pages 268–278. IEEE, 1995.