Lecture 2 in "Robust computation: from local pieces to global structure"

Irit Dinur

August 8, 2025

1 Robust characterization, systems of constraints

We begin with some formal definitions, putting our discussion from last week into a more rigorous framework.

Notation and definitions

A system of constraints is given by a hypergraph $H = (V, E)$, an alphabet Σ, and a constraint $C_e \subseteq \{f : e \to \Sigma\}$ for each hyperedge $e \in E$. The constraint describes which assignments to the vertices $v \in e$ are allowed and which are not.

An assignment $f : V \to \Sigma$ satisfies the constraint C_e iff $f|_e \in C_e$. An assignment f satisfies the system of constraints if it satisfies all of the constraints, i.e., $f|_e \in C_e$ for all $e \in E$. We refer to the entire system of constraints as H and denote the set of satisfying assignments of H by

$$SAT(H) = \{f : V \to \Sigma \mid \ f|_e \in C_e, \ \forall e \in E\}.$$

We say that $SAT(H)$ is *characterized* by H. What about robustness?

For a given assignment f, it is not clear easy to tell how close it is to the set $SAT(H)$, but much easier to measure the fraction of constraints that it satisfies,

$$val(f) = \frac{\sum_{e \in E} \mathbf{1}_{C_e}(f|_e)}{|E|} = \mathop{\mathbb{P}}_{e \in E}[f|_e \in C_e]$$

Similarly $rej(f) = 1 - val(f)$ is the fraction of constraints that f does not satisfy, i.e., the fraction of constraints that it rejects. We define

$$val(H) = \max_{f:V \to \Sigma} val(f), \quad rej(H) = 1 - val(H) \tag{1.1}$$

A system is called *c-robust* if for any assignment f, $rej(f)$ is a good measure for the distance of f from a satisfying assignment.

Definition 1.1 (Robustness)**.** Given a system of constraints H, the robustness of H is defined as

$$c = \min_{f \notin SAT(H)} \frac{rej(f)}{\mathrm{dist}(f, SAT(H))}.$$

In words, c is the largest real number such that for every assignment $f : V \to \Sigma$, $rej(f) \geqslant c \cdot \mathrm{dist}(f, SAT(H))$.

In this terminology, we have proven in the previous lecture that the system of linearity testing equations is a robust system of constraints, with $c = 4/9$. Indeed, if $rej(f) \geqslant 2/9$ this is trivial because $\frac{4}{9}\,\mathrm{dist}(f, SAT(H)) \leqslant \frac{4}{9} \cdot \frac{1}{2} = \frac{2}{9} \leqslant rej(f)$ (since for any f it is $1/2$ close to either all 0 or all 1), and if $rej(f) < 2/9$, we saw that $\frac{2}{3}\,\mathrm{dist}(f, SAT(H)) \leqslant rej(f)$ and thus the system is robust with $c = \min(2/3, 4/9) = 4/9$.

Locally testable codes. A linear code C is locally testable with q queries and robust soundness parameter ρ if there exists a ρ-robust system of (linear) constraints H that characterizes C, namely such that $C = SAT(H)$. Moreover, each constraint in H involves no more than q variables.

The condition $C = SAT(H)$ is sometimes called perfect completeness since it means that for every $f \in C$, $rej(f) = 0$. The condition of ρ-robustness is related to soundness because if f is δ-far from C then $rej(f) > \rho \cdot \delta$.

Relation to property testing. In the area of property testing, the focus is on a property, say $P \subset \mathbb{F}_2^n$, whether or not there is a tester for it, and with how many queries. The tester (at least in the non-adaptive case) can be viewed as a robust system constraints (each constraint is defined by what the tester looks at for a fixed choice of the randomness, and which views cause it to accept). For example, a locally testable code is a code C and if there is a tester for it, this can translate directly to the existence of a robust system of constraints[1]. In the formulation above, the emphasis, or the focus, is more on the system of constraints, compared to caring mainly about the codewords (or more generally the property, $SAT(H)$).

Coboundary expansion. We will see later on a definition of a linear map from the assignment to the constraints $\delta : \mathbb{F}_2^V \to \mathbb{F}_2^E$ called the coboundary map δ. This map takes an assignment $f : V \to \mathbb{F}_2$ to the set of constraints it violates, $\delta(f)_e = \begin{cases} 0 & f|_e \in C_e \\ 1 & \text{otherwise} \end{cases}$. This is a linear map if the constraints

[1] I am ignoring the case of non-perfect completeness.

are linear, and robustness of H becomes exactly the coboundary expansion of this map.

2 Low degree tests

We now turn to another set of functions, of potentially much higher density, that is also characterized by a robust system of constraints, namely the set of low degree polynomials.

Let q be a prime power. A polynomial $f : \mathbb{F}_q^m \to \mathbb{F}$ has total degree d if

$$f(x) = \sum_{(e_1,\ldots,e_m),\sum e_i \leqslant d} a_e \prod_{i=1}^{m} x_i^{e_i}$$

The set of polynomials of degree at most d is denoted by $RM(m,d)$ and is called the Reed-Muller code. It is a linear code, and one can calculate its dimension to be $|RM(m,d)| = \binom{m+d}{d}$. The relative distance is $1-d/q$.

2.1 Characterization of low degree polynomials

What kind of equations does a polynomial of degree at most d satisfy? Assume that $q > d+2$. When $m = 1$ we know that any $d+1$ points $x_1,\ldots,x_{d+1}$ and any $d+1$ values $y_1,\ldots,y_{d+1}$ determine uniquely a univariate polynomial f of degree at most d such that $f(x_i) = y_i$. In fact, this gives a robust test: choose at random $x_0,\ldots,x_{d+1}$ and accept if $f|_{\{x_0,\ldots,x_{d+1}\}}$ agrees with some degree-d polynomial. Clearly this will always succeed in case $f \in RM(1,d)$. Moreover, denoting $agr = 1 - dist$,

Claim 2.1. If $Prob_{x_0m\ldots,x_{d+1}}[f|_{\{x_0,\ldots,x_{d+1}\}}$ agrees with some degree-d polynomial$] = \alpha$, then $agr(f, RM(1,d)) \geqslant \alpha$

Proof. Assume that the test passes with probability α. There must be some $x_1,\ldots,x_{d+1}$ such that the test passes with probability α even conditioned on $x_1,\ldots,x_{d+1}$. Let g be the univariate polynomial of degree at most d that agrees with f on these points. Then g agrees with f on α fraction of the remaining points in $\mathbb{F}\setminus\{x_1,\ldots,x_{d+1}\}$, so altogether $agr(f, RM(1,d)) \geqslant agr(f,g) \geqslant \alpha$. ◻

Moving to $m = 2$, how would we test bivariate polynomials? If we choose random $d+2$ points, there might not be any relation between them that we can check. It is natural to look at the restriction of f to a random axis-parallel line, say $f(\cdot,a)$ or $f(a,\cdot)$. This is a good test, and a nice analysis was given by Polyschuk and Spielman [PS94]. How does this test generalize to larger m? The so-called “axis-parallel line test” will choose a random $i \in [m]$ and a random point $a \in \mathbb{F}^m$ and then look at $f(a_1, \ldots, a_{i-1}, \cdot, a_{i+1}, \ldots, a_m)$, namely at a random axis parallel line. When m grows the robustness will decrease proportionally to $1/m$ as can be seen from the function $f(x) = (x_i)^{d+1}$. This polynomial is far from any degree d function (because for any polynomial of degree $\leqslant d$, the difference is a non zero polynomial of degree $d+1$, and it can have no more than $\frac{d+1}{q}$ fraction of zeros by the Schwartz-Zippel lemma), yet it passes the axis-parallel line test with probability $1 - 1/m$.

The dependence of the robustness on m can be removed with the following test:

– Choose a random $x \in \mathbb{F}_q^m$ and a random $h \in \mathbb{F}_q^m$ such that $h \neq 0$. Let $\ell_{x,h} = \{x + ih \mid i \in \mathbb{F}\}$.

– Read $f|_{\ell_{x,h}}$ and check if it agrees with some degree-d polynomial on this line.

In fact, the second step can be replaced by reading f at a random set of $d+2$ points on the line $\ell_{x,h}$, and checking if these values agree with some degree-d polynomial. This is due to Claim 2.1.

This test (actually, a variant of it) was analyzed by Rubinfeld and Sudan [RS96]. It is quite similar to the analysis of linearity testing. It proceeds by defining a self-corrected function g (by plurality vote) and then showing that (a) g is close to f, (b) The plurality vote is by high margin, and then (c) g must be low degree. The last two steps involve using some nice dependencies between the constraints of the test, namely the fact that an arbitrary constraint can be expressed as a short sum of other (more random) constraints.

3 Line versus point test and other agreement tests

The set of functions $RM(m, d)$ has polynomial density inside the set of all functions $\{f : \mathbb{F}^m \to \mathbb{F}\}$ when $d \approx m \approx \log(\binom{m+d}{d})$. In this case the low degree test makes a logarithmic number of queries (since $d \approx m = \log \mathbb{F}^m$). Can the number of queries be reduced further?

One idea is to enhance the input, by adding in addition to $f : \mathbb{F}^m \to \mathbb{F}$ another piece of encoding, called the lines table (or lines oracle), which supposedly gives the restriction of the function f to all possible lines. The lines table is a collection $\{f_\ell\}_\ell$, where ℓ is an affine line and $f_\ell : \ell \to \mathbb{F}$ is a univariate degree d polynomial (given, for example, through $d+1$ coefficients). In the lines table, the intent is that $f_\ell = f|_\ell$. Namely, in a valid encoding, f has degree d, and each f_ℓ is its restriction to the line ℓ. Now we can use the collection $\{f_\ell\}$ to help us test if f is low degree, keeping in mind that there is no apriori guarantee that f_ℓ are consistent with each other or with a global low degree function.

Given both f and $\{f_\ell\}$, a natural *test* that this is a representation of a low degree function is as follows

Line vs. point test.

- Choose a random $x \in \mathbb{F}^m$ and a random line $\ell \ni x$.
- Accept if $f(x) = f_\ell(x)$.

The following lemma shows that analyzing the line vs point test loses no generality compared to the basic low degree test.

Lemma 3.1. *Given $f : \mathbb{F}^m \to \mathbb{F}$ that passes the basic low degree test with probability α, there is a lines table $\{f_\ell\}_\ell$ such the pair $f, \{f_\ell\}$ pass the line vs. point test passes with probability at least α as well.*

Proof. Given $f : \mathbb{F}^m \to \mathbb{F}$ that passes the basic low degree test with probability α, we can construct a lines table $\{f_\ell\}_\ell$ as follows. For each line ℓ, let f_ℓ be the degree d polynomial that agrees with f on the maximal number of points in ℓ. Since f passes the basic low degree test with probability α, it follows that for a random line ℓ, the restriction $f|_\ell$ will be at least α-close to a degree d polynomial (see Claim 2.1), on average. Therefore, the pair $f, \{f_\ell\}$ will pass the line vs. point test with probability at least α as well. □

This test has been analyzed by Arora and Sudan [AS03]. We will describe another test, which was analyzed concurrently by Raz and Safra [RS97] and whose analysis is more combinatorial. For this test, we ask for the collection of restrictions of f to planes, not lines.

Plane vs. plane test. Input: $\{f_s \mid \ f_s : s \to \mathbb{F} \text{ is bivariate with degree at most } d\}$ where s ranges over all possible affine planes in $\mathbb{F}^m$.

- Choose a random line ℓ, and two random planes $s, s' \supset \ell$.
- Accept if $f_s|_\ell = f_{s'}|_\ell$.

The analysis begins by looking at the case $m = 3$.

4 Analysis of the plane vs. plane test

Let us consider the *consistency graph* of the test, which is a graph whose vertices are the planes, and where we put an edge between s, s' if $f_s|_\ell = f_{s'}|_\ell$, where ℓ is the intersection line. Observe that since $m = 3$ every pair of distinct planes intersect in a line or are parallel. If s, s' are parallel we will also put an edge between s, s'. We will write $s \sim s'$ to denote that there is an edge between s, s'. By assumption,

$$\alpha = \mathop{\mathbb{P}}_{s,s'}[s \sim s'].$$

The key is the following structural restriction on the edges and non edges in the consistency graph.

Claim 4.1. Let s, s' be two planes in $\mathbb{F}^3$ such that $s \nsim s'$. At most $\frac{d+1}{q}$ of the planes s'' have $s'' \sim s$ and $s'' \sim s'$. We call such triples $\{s, s', s''\}$ *bad triangles.*

Proof. If $s \nsim s'$, then there are at most d point on ℓ such that $f_s(p) = f_{s'}(p)$. Choose a random s''. With probability $1/q$, s'' is parallel to ℓ (namely, either disjoint from ℓ or contains it). With the remaining probability, it must intersect ℓ at a point. So with all but $\frac{1}{q} + \frac{d}{q}$ probability, s'' intersects $\ell = s \cap s'$ on a point p such that $f_s(p) \neq f_{s'}(p)$ and so either $f_{s''}(p) \neq f_{s'}(p)$ or $f_{s''}(p) \neq f_s(p)$ (or both). This means that s'' cannot be adjacent to both s and s', and thus there are at most $\varepsilon = \frac{d+1}{q}$ planes that are adjacent to both s and s'. □

For a vertex v, let ε_v be the fraction of edges uw such that $u \sim v$, $w \sim v$ but $u \nsim w$.

Claim 4.2. $\mathbb{E}_v[\varepsilon_v] \leqslant \varepsilon := \frac{d+1}{q}$.

Proof. Consider the bipartite graph between V and E, where we connect a vertex v to an edge uw if $u \nsim w$ yet $u, w \sim v$. Every non edge $u \nsim w$ has degree at most $\frac{d+1}{q}|V|$ according to the previous claim. Averaging from the vertex side we get that the average degree of a vertex is $\mathbb{E}[\varepsilon_v|E|] \leqslant \frac{d+1}{q}|E|$. □

Claim 4.3. There must be a vertex v^* with at least $(\alpha - 2\sqrt{\varepsilon})|V|$ consistent vertices $u \sim v^*$, and such that $\varepsilon_{v^*} \leqslant \sqrt{\varepsilon}$.

Proof. There cannot be more than $\sqrt{\varepsilon}$ vertices with $\varepsilon_v \geqslant \sqrt{\varepsilon}$, by averaging. Of those that remain, choose a vertex that agrees with a maximal number of vertices u. It must agree with at least $\alpha - 2\sqrt{\varepsilon}$ (because the vertices with high ε_v have been removed, and even if each was consistent with all other vertices, they could only contribute $2\sqrt{\varepsilon}$ to the total agreement, which now decreases from α to $\alpha - 2\sqrt{\varepsilon}$). □

Let $A \subset V$ be the set of planes that are consistent with v^*, so $|A| = (\alpha - 2\sqrt{\varepsilon})|V|$. By our choice of v^*, $\varepsilon_{v^*} \leqslant \sqrt{\varepsilon}$, so there are relatively few non-edges inside A.

Claim 4.4. Let $\beta = (\alpha - 2\sqrt{\varepsilon} - \varepsilon)/2$, and let $B \subset A$ be the set of planes that are inconsistent with at least $\beta|V|$ planes in A. Then $A \setminus B$ is a clique in the consistency graph, and $|A \setminus B| \geqslant (\alpha - \frac{\varepsilon}{\beta})|V|$.

Proof. For every $u, w \in A$, if $u \nsim w$ then by Claim 4.1, there are at most $\varepsilon|V|$ planes that are consistent with both u and w. This means that the remaining $r \in A$ have either $u \nsim r$ or $w \nsim r$, so one of u, w must have at least $(|A| - \varepsilon|V|)/2 = \beta|V|$ non-neighbors inside A, so fall into B. Thus, $A \setminus B$ is a clique. □

Finally, let us bound the size of the set B. Each $r \in B$ touches $\beta|V|$ non-edges, which make at least $\beta|V|$ bad triangles involving v^*, while the total number of those is $\varepsilon_{v^*}|E| \leqslant \sqrt{\varepsilon}|E|$. Each bad triangle can be counted at most twice, so we get that $|B| \cdot \beta|V|/2 \leqslant \sqrt{\varepsilon}|E|$, and thus $|B| \leqslant \frac{2\sqrt{\varepsilon}|E|}{\beta|V|} = \frac{\sqrt{\varepsilon}}{(\alpha - 2\sqrt{\varepsilon} - \varepsilon)}|V|$.

This analysis gives a good bound when $\alpha \gg \sqrt{\varepsilon}$. For example if $\alpha > 1/q^{1/4}$ then the clique has size at least $\alpha - 1/\sqrt{q}$. This assumption on α is not needed in the original Raz-Safra proof [RS97].

References

[AS03] Sanjeev Arora and Madhu Sudan. Improved low-degree testing and its applications. *Comb.*, 23(3):365–426, 2003. 5

[PS94] A. Polishchuk and D. Spielman. Nearly linear size holographic proofs. In *Proc. 26th ACM Symp. on Theory of Computing*, pages 194–203, 1994. 3

[RS96] Ronitt Rubinfeld and Madhu Sudan. Robust characterizations of polynomials with applications to program testing. *SIAM Journal on Computing*, 25(2):252–271, 1996. 4

[RS97] R. Raz and S. Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In *Proc. 29th ACM Symp. on Theory of Computing*, pages 475–484, 1997. 5, 7