

# PCP - Huji Course, Ex 2.

David Ponnarovsky

July 5, 2023

## 1 Ex 1. Sumchecking with coefficients.

We would like to verify that a given polynomial box  $P$  satisfies that  $\sum_{x \in [d]^m} \varphi(x) f_P(x) = 0$  by accessing to at most  $O(md)$  variables. For any function  $\varphi : [d]^m \rightarrow \mathbb{F}_q$ . Denote by  $\varphi' : \mathbb{F}_q^m \rightarrow \mathbb{F}_q$  the extension of  $\varphi$  into a polynomial over  $\mathbb{F}_q^m$ . We saw in that lectures (and also in the previous assignment) that there is such a unique extension.

We are going to split the section into three, first we are going to show how to verify that  $\sum_{x \in [d]^m} f_P(x) = 0$ . When the polynomial is a function into  $\mathbb{F}_q$ . (I think, but not sure, that in the lecture we saw only the case when  $q = 2$ ). Then in the second part we will show how can one reduce the coefficients case into the non-coefficients case. Finally, in the last part, we combine all together to show that the construction achieves the requirements.

### 1.1 Over non binary field.

Let's define a series of polynomial boxes  $f_i$  such that:

$$f_0 = f$$
$$f_{i+1}(x_1, \dots, x_{m-i}) = \sum_{y \in [d]} f_i(x_1, \dots, x_{m-i}, x_{m-i+1} = y)$$

Our verifier will ask for a proof which is a list of  $f_0, f_1, f_2, \dots, f_m$ . Now, notice that if  $f$  is an honest assignment then  $f_m$  is just the summation of  $f$  over the cube  $[d]^m$ . So it is sufficient to show the existence of a verifier that rejects with high probability any string far from being encoded by the previous structure.

- 1 Sample uniformly random  $i \sim [m]$  and check that  $f_i$  is a codeword of the polynomial code in  $m$  variables at degree at most  $m \cdot d$ .
- 2  $r_1, r_2, \dots, r_m \leftarrow$  sample uniformly  $m$  points of  $[d]$
- 3 **for**  $i \in [1, m]$  **do**
- 4     Check if  $f_{i+1}(r_1, \dots, r_{m-i-1}, x_{m-i}) - \sum_{y \in [d]} f_i(r_1, r_2, \dots, r_{m-i-1}, x_{m-i}, y)$  is the zero polynomial by a random test that uses at most single query. (Here  $x_{m-i} \in [d]$  is the only variable)
- 5
- 6     If not then reject.
- 7 **end**
- 8 Accept if  $f_0 = 0$

*Proof.* For convenience let's denote by  $g_i(x_{m-i})$  the difference that has been queried in line number 3.

1. Correctness. Easy. If the assignment is honest then by definition  $g_i = 0$  for any  $i \in [m]$  and therefore for any  $x_{m-i}$  we will have that  $g_i(x_{m-i}) = 0$ . So, in that case iteration will pass. And whole proof will be accepted with probability 1.
2. Soundness. Assume that there is any  $\deg f_{i+1} \leq \deg f_i$ , Thus asking

□

## 1.2 Coefficients $\mapsto$ non-coefficients.

Now as we proved in the classes  $\deg f \cdot g \leq \deg f + \deg g$ . Therefore we can redact the problem of verifying whether the weight summation is zero by considering the summation of the polynomial  $\varphi' \cdot f$  over the cube  $[d]^m$ .

- 1 Sample uniformly random  $x \sim [d]^m$  and check that  $\varphi'(x) = \varphi(x)$
- 2 Check that  $\varphi$  is a polynomial at degree at most  $d \cdot m$ .
- 3 If both of the checks passed, accept.

*Proof.* □

## 1.3 Combine all.

- 1 Use the first tester to check the validity of the pair  $(\varphi', \varphi)$ .
- 2 Check that the degree of  $\xi$  is at most  $2md$
- 3 Check that the polynomial  $f \cdot \varphi' - \xi$  is the zero polynomial.
- 4 Using the first verifier, accept if the summation of  $\xi$  over the cube  $[d]^m$  is zero.

*Proof.* □

## 2 Ex 2.

The question concerns with the following test: [\[COMMENT\] rewrite again.](#)

- 1 Choose  $x, y \in \{\pm 1\}^k$  independently.
- 2 Choose  $\mu \in \{\pm 1\}$ .
- 3 Choose a random noise  $z \in \{\pm 1\}^k$  such that  $z_i$  gets +1 with probability  $1 - \varepsilon$ .
- 4 Accept if  $\mu f(\mu x) \cdot g(y) = f(z \cdot xc^{-1}(y))$

### 2.1 2.a.

Let  $f = \chi_{\{i\}}, g = \chi_{\{j\}}$  and  $j = c(i)$ . In that case it holds that:

$$\begin{aligned} \mu f(\mu x) \cdot g(y) &= \mu \chi_{\{i\}}(\mu x) \chi_{\{j\}}(y) = \mu^2 x_i y_j = x_i y_j \\ f(z \cdot xc^{-1}(y)) &= \chi_{\{i\}}(zxc^{-1}(y)) = z_i x_i y_j \end{aligned}$$

Thus, the test pass only if  $z_i = 1$  and it given that this event happens with probability  $1 - \varepsilon$ .

### 2.2 2.b.

Denote by  $\alpha_I \in \mathbb{R}$  and  $\beta_J \in \mathbb{R}$  the coefficients of  $f, g$  over the character  $\chi_{\{I\}}$ .

$$\begin{aligned} &\mathbf{E} [\mu f(\mu x) \cdot g(y) f(z \cdot xc^{-1}(y))] \\ &= \sum_{I, J, K} \alpha_I \alpha_K \beta_J \mathbf{E} [\mu \chi_{\{I\}}(\mu x) \chi_{\{J\}}(y) \chi_{\{K\}}(zxc^{-1}(y))] \\ &= \sum_{I, J, K} \alpha_I \alpha_K \beta_J \mu^{|I|+1} \mathbf{E} [\chi_{\{I\}}(x) \chi_{\{J\}}(y) \varepsilon^{|z|} (-1)^{|z \cap K|} \chi_{\{K\}}(xc^{-1}(y))] \end{aligned}$$

So it left to compute the expectation  $\mathbf{E} [\chi_{\{I\}}(x) \chi_{\{J\}}(y) \chi_{\{K\}}(xc^{-1}(y))]$  and observes that if  $c^{-1}(y)$  has no intersection with  $K$ . Define by  $C(K)$  all the indices  $i$  such that there exist  $k \in K$  for which  $y_{c_k} = y_i$ .

$$\mathbf{E} \left[ \prod_{i \in I} x_i \prod_{j \in J} y_j \prod_{k \in K} x_k \cdot y_{c_k} \right] = \mathbf{E} \left[ \prod_{i \in I \Delta K} x_i \prod_{j \in J \Delta C(K)} y_j \right]$$

*Proof.*

□

### 2.3 Ex 3. The label cover problem.

Let us assume that that  $|\Sigma|$  is a power of 2. Associate for each vertex a vector in  $\mathbb{F}_2^{|\Sigma|}$  (Soon we will add more  $\Theta(|\Sigma|)$  variables for having a sparse sum checking, namely for checking that  $\sum_j^{|\Sigma|} x_{vj} = 1$ ). And for each constraint.

Idea, there are more than  $\mu < ++ >$  equations that satisfied then  $\Rightarrow$  there are more than  $\Theta(|V|)$  which their local environment is  $\frac{1}{2}\mu$  satisfied.  $\Rightarrow$  the local  $T_\varepsilon(c)$  test accepts with probability  $\frac{1}{2} + \delta(\mu)$  and therefore there exist  $i \in L_\delta(f), j \in M_\delta(g)$  s.t  $c(i) = j$ .  $\Rightarrow$  we could pick  $f$  and  $g$  to be  $\chi_{\{i\}}$  on those vertices and get a solution such  $(1 - \varepsilon) \cdot < ++ >$  are satisfied.

Other direction to consider, suppose that we satisfy more than  $\frac{1}{2} + \delta$  equations, than for at least  $\Theta()$  of the edges we success to find  $i, j, i = c(j)$ . Therefore we can construct another assignment in which at least  $1 - \varepsilon$  of the equations are satisfied.

## 3 Part 3.

**Label cover when the aleph-bet depends on the vertex.** Instead of showing reduction into the general label cover we will show a reduction to a similar problem in which vertices can have an additional restriction on the valid characters that one can sets on. In formal, we will say that  $\langle G, \{\Sigma_v : v \in V\}, \{c_e : e \in E\} \rangle$  instance of Generalized-Label-Cover if there is an assignment  $A : V \rightarrow \Sigma$  such that for any  $\{v, u\} \in E$  it holds that  $c_e A(v) = A(u)$  and in addition for any  $v \in V$  we have that  $A(v) \in \Sigma_v \subset \Sigma$ .

**The reduction.** Define the Bipartite graph  $G = (L, R, E)$ . Associate the left vertices with the variables and the right with the closures. Define  $\{u, v\}$  to be an edge if the literal which associate with the vertex  $u$  is in the closure associate with vertex  $v$ . For the alphabet take  $\Sigma = \mathbb{Z}_2^3$ . For any right vertex  $v \in R$  define  $\Sigma_v$  be all the assignments for which the  $v$ -closures is satisfied. Finally define  $c_e$  for  $e = \{v \in R, u \in L\}$  to be the projection of  $\sigma \in \Sigma$ , setted on  $v$ , to the coordinate corresponding with  $u$ . For example, assume that  $v$  associate with  $x \vee y \vee z$  and let  $u$  be the vertex associate with  $x$ , And assume that  $A(v) = (1, 0, 1)$ , then  $c_e A_v = (1, 0, 0)$ .

**Completnce.** Suppose that  $\varphi \in \text{E3-CNF-SAT}$  and let  $x \in \mathbb{F}_2^*$  be the assignment that satisfies  $\varphi$ . That it,  $\varphi(x) = \text{True}$ . Let  $A$  be the assignment that sets for any vertex on the left the bit matched to that literal by  $x$  follows by zeros padding. And for any right vertex the triple of the bits corresponding to literals involving in the associated closure.