6.841 Advanced Complexity Theory	April 25, 2007
Lecture 20	
Lecturer: Madhu Sudan	*Scribe: Alex Schwendner*

1 Generalized r-Graph k-coloring

Our goal for this lecture is to prove the PCP Theorem using Dinur's proof by Gap Amplification [Din06]. For this, we'll be using our second view of PCPs, namely Generalized Graph Coloring.

For a given triple (k, r, ϵ), the Generalized r-Graph k-coloring problem consists of a graph (V, E) along with a notion of which colorings are valid for each hyper-edge:

$$G = (V, E, \text{valid} : E \times [k]^r \longrightarrow \{0, 1\})$$
$$E \subseteq \underbrace{V \times V \times \cdots \times V}_{r\text{-times}}$$

We define a *k-coloring* to be an assignment of one of k colors to each vertex: $\chi : V \longrightarrow \{1, \ldots, k\}$. For a coloring χ, we call an edge $e = (v_1, \ldots, v_r)$ *satisfied* if $\text{valid}(e, \chi(v_1), \ldots, \chi(v_r)) = 1$. The graph G is then called *k-colorable* if there exists a coloring χ such that all edges are satisfied.

We consider the promise problem for which we want to accept if G is k-colorable, and reject if at least an ϵ fraction of the edges are unsatisfied in any coloring of G. If we define

$$\text{unsat}_\chi(G) = \frac{|\{e \in E \mid e \text{ not satisfied by } \chi\}}{|E|}$$
$$\text{unsat}(G) = \min_\chi \{\text{unsat}_\chi(G)\}$$

then we can specify that we wish to reject G if $\text{unsat}(G) \geq \epsilon$.

2 Reductions

2.1 Definition

We define a reduction $(k, r, \epsilon) \longrightarrow (k', r', \epsilon')$ to be a linear-time reduction mapping an r-graph G to an r'-graph G' such that if G is k-colorable, then G' is k'-colorable, and if $\text{unsat}(G) \geq \epsilon$ then $\text{unsat}\, G' \geq \epsilon'$:

$$\begin{aligned} r\text{-graph } G &\longrightarrow r'\text{-graph } G' \\ G\ k\text{-colorable} &\Longrightarrow G'\ k'\text{-colorable} \\ \text{unsat}(G) \geq \epsilon &\Longrightarrow \text{unsat}(G') \geq \epsilon' \end{aligned}$$

2.2 Classical Reductions

Here are some easy classical (or neo-classical) reductions:

1. $(k, r, \epsilon) \longrightarrow (2, r \log k, \epsilon)$ — Here, we replace each vertex with $\log k$ vertices with binary colors that encode the previous $(\log k)$-bit coloring. Then, we expand each edge to cover all $r \log k$ vertices used to represent the former r vertices.
2. $(k, r, \epsilon) \longrightarrow \left(k^r, 2, \frac{\epsilon}{r}\right)$ — This was in Lecture 18.
3. $(k, 2, \epsilon) \longrightarrow \left(3, 2, \frac{\epsilon}{f(k)}\right)$ — [PY88]
4. $(2, r, \epsilon) \longrightarrow \left(2, 3, \frac{\epsilon}{r}\right)$ — By Cook
5. $(k, r, \epsilon) \longrightarrow (k, c \cdot r, \underbrace{\approx \epsilon \cdot c}_{=1-(1-\epsilon)^c})$ — based on expander walks

However, these don't help us prove the PCP Theorem. In reductions 1 through 4, ϵ only goes down, but we need to make ϵ equal to $1/2$ to prove the PCP theorem. In reduction 5, ϵ does get bigger, but the ratio $\frac{r \cdot \log k}{\epsilon}$ still gets bigger, not smaller. This ratio is approximately what we want to have be small, but none of these reductions decrease it.

3 Dinur

The reductions in 2.2 aren't enough to prove the PCP Theorem. Dinur's proof, however, relies on two key lemmas.

Lemma 1 *(Gap Amplification)* $\forall\, c, k\ \ \exists\, K$ *such that* $\forall\, \epsilon$,

$$(k, 2, \epsilon) \longrightarrow (K, 2, c \cdot \epsilon).$$

Lemma 2 $\exists\, \delta$ *such that* $\forall\, K$,

$$(K, 2, \epsilon) \longrightarrow (2, 4, \epsilon\delta).$$

Note that the δ in Lemma 2 is fixed. Thus, fix $c = 8/\delta$. Now, for $k = 16$, let K be as implied by Lemma 1. We can combine Lemma 1 and Lemma 2 with Reduction 2 from Section 2.2 to prove

Lemma 3 $(16, 2, \epsilon) \longrightarrow (16, 2, 2\epsilon)$:

$$\begin{aligned}(16, 2, \epsilon) &\xrightarrow{\text{Lemma 1}} (K, 2, \epsilon c) \\ &\xrightarrow{\text{Lemma 2}} (2, 4, 8\epsilon) \\ &\xrightarrow{\text{Reduction 2}} (16, 2, 2\epsilon)\end{aligned}$$

What Lemma 3 lets us do is increase the size of the gap from ϵ to 2ϵ with a linear reduction.

We claim that our work in Lecture 19 can be seen to imply Lemma 2. Informally, a PCP is more than just a proof, but is a commitment to a specific proof. We can adapt our with in showing how to check a PCP to check a graph coloring.

Let $l = \log k$. We split each vertex v into a cloud of k vertices. For each of these vertices i, we choose a linear function L_i and let the color of i be the bit $L_i(\chi(v))$ where $\chi(v)$ is considered as a $(\log k)$-bit vector. The constraints that we checked of the form $\Pi[Q_1] + \Pi[Q_2] = \Pi[Q_1 + Q_2]$ for a valid PCP get modeled as 3-edges.

Next, Dinur's proof of Lemma 1. We first sketch a weak reduction with expander walks as in Reduction 5. There, we take a random walk on G, and take the conjunction of the constraints on the edges we traverse. If G is an expander then this will amplify the error gap (ϵ). However, with this, r will increase but we need r to stay the same while k increases to K.

Instead, in Dinur's construction, we start by fixing a constant t and then letting $B_v = B_v^t = \{u \mid \delta(u,v) \leq t\}$ where $\delta(u,v)$ is the length of the shortest path between u and v in G. We now define our reduction

$$G = (V, E, \text{valid})$$
$$\downarrow$$
$$G' = G'_t = (V', E', \text{valid}')$$

where

$$V' = V$$
$$E' = \left\{(u,v) \mid \exists w_1 \cdots w_l \text{ s.t. } w_1 = u, w_l = v, (w_i, w_{i+1}) \in E, \frac{t}{2} \leq l \leq t\right\} \text{ as a multiset}$$
$$\chi' \colon u \longmapsto \{\chi_u \colon B_u \longrightarrow \{1, \ldots, k\}\}.$$

That is, each new edge (u,v) is the collection of paths from u to v, and each new coloring of u $\chi'(u)$ is a function specifying the old coloring on B_u, the neighborhood of u. We then require that these colorings can be stitched together to form a valid coloring of the entire graph.

Specifically, for some u and v connected by an edge $(u,v) \in E'$, $\text{valid}'(\chi_u, \chi_v) = 1$ if

1. $\forall\, w \in B_u \cap B_v,\ \chi_u(w) = \chi_v(w)$
2. $\forall e = (w_1, w_2) \in E$ with $w_1, w_2 \in B_u \cap B_v$, $\text{valid}(e, \chi_u(w_1), \chi_v(w_2)) = 1$.

This construction produces a graph with r still equal to 2. However, our new colors imply much more about the state of the graph and thus checking each edge gives more information. By choosing a large enough value for t, we can then get any desired increase in strictness ϵ. The number of colors increases to some K. This proves Lemma 1.

References

[Din06] Irit Dinur. The pcp theorem by gap amplification. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 241–250, New York, NY, USA, 2006. ACM Press.

[PY88] Christos Papadimitriou and Mihalis Yannakakis. 1988.