Smoothed Complexity of Local Search of Max-Cut for graph with logarithmic degree

October 2, 2017

1 Introduction

Let $G(V, E)$ be a graph with a set V of n vertices and E of m edges. Forall vertex v that belongs to V, the degree of v is inferior or equal to $c \cdot log(n)$ where c is a constant. Let $w : E \rightarrow]-1, 1[$ be an edge weight function. This restriction is for commodity in the proof, we can easily generalize to other weight function by dividing all weight by $w_{max} = max_e(|w_e|)$ as long as the variance of the weight function is bounded.
Let $\sigma : V \rightarrow -1, 1$ be a partition and let the size of the cut corresponding to this partition be :

$$\frac{1}{2}\sum_{uv \in E} w(uv)(1 - \sigma(u)\sigma(v))$$

We call a cut locally optimal, if there exists no vertex v such that if v is moved to the other side of the cut, the size of the cut increases. A simple algorithm called FLIP solves the problem of finding a local-optimal cut. Search for a vertex which, if flipped, improves the size of the cut (we call it an improving move), if there is none stop, else repeat. Let call s such a sequence of move

Formally we now assume that the weight on edge $e \in E$ is given by a random variable $X_e \in [-1, 1]$ which has a density with respect to the Lebesgue measure bounded from above by ϕ (for example this forbids X_e to be too close to a point mass). We assume that these random variables are independent.

We will prove here that the expected number of step is polynomial with high probability, i.e. the smoothed complexity is polynomial.

2 Proof

2.1 Bounding a move

Let call t a move. Remark that α_t, the amelioration to the cut probided by the move t, depends only on the vertex which is flipped and its neighbors. Here we use a particular case of the lemma 7 provided by Etscheid and Rglin[2014].
For $\epsilon > 0$ and a fixed move t $\mathbf{Pr}(\alpha_t \in [0, \epsilon]) \leq \phi\epsilon$

2.2 Using union bound

We use the union bound over the n possible vertices and the $2^{c \cdot log(n)}$ possible configuration of their neighbors. This gives us that for any move t :
$\mathbf{Pr}(\alpha_t \in [0, \epsilon]) \leq n2^{clog(n)}\phi\epsilon$

2.3 Finding a good epsilon

Now we choose $\epsilon = \dfrac{\delta}{n2^{clog(n)}\phi}$ With some $\delta \in]0, 1[$ Thus :

$$\mathbf{Pr}(\alpha_t \in [0, \epsilon]) \leq \delta \implies \mathbf{Pr}(\alpha_t \geq \epsilon) \geq 1 - \delta \tag{1}$$

Where we used that a move is always an improvement over the size of the cut.

2.4 Using Bernouilli trials

Since a cut is in $[-c \cdot nlog(n), c \cdot nlog(n)]$, the maximum improvement done by all the moves before the algorithm stop is at most $2c \cdot nlog(n)$.
Let call T the set of the moves in s which bring an improvement of at least ϵ, then :
$|T| \leq \dfrac{2c \cdot nlog(n)}{\epsilon}$

Let X be the expected number of steps, we can bound X by the expected number of Bernoulli trials with probability $p = 1 - \delta$ needed before having $k = \dfrac{2c \cdot nlog(n)}{\epsilon}$ success.

$$\begin{aligned} X \leq & \frac{k}{p} \\ X \leq & \frac{n2^{c \cdot log(n)} 2c \cdot nlog(n)\phi}{(1-\delta)\delta} \\ X \leq & \frac{2c \cdot n^{c+2} log(n)\phi}{(1-\delta)\delta} \end{aligned} \tag{2}$$

We can see that X is $O(n^{c+3}\phi)$, thus the complexity is polynomial in n and ϕ

But is it really it ? Or am I proving something weaker ?

Is there a choice of ϵ such that $(1 - n^{c+1}\phi\epsilon)^{\frac{2cnlog(n)}{\epsilon}}$ is o(1) ?