Sublinear Algorithms	September 22, 2015

Lecture 25

Lecturer: Sofya Raskhodnikova *Scribe(s): Roksana Baleshzar*

Today, we will discuss an algorithm for approximating the average degree of a graph. This algorithm is from a paper by Oded Goldreich and Dan RonGoldreich and Ron [1].

1 Types of Queries

In this problem we suppose that we have access to two types of queries like the following:

1. Degree queries: Which queries of the form (v) and give the degree of the vertex v in the output.
2. Neighbour queries: That is, queries of the form (v, i) that are give the i^{th} neighbour of the vertex v as the output.

We can also define a new type of queries which get a vertex (v) and output a random neighbour of v, it is clear that this query implemented from two previous queries.

2 Difficulty of Multiplicative Approximation

We define the $k-$multiplicative $\hat{a}$ for the actual answer a such that $\frac{a}{k} \leq \hat{a} \leq k.a$. In this section we are going to see some difficulties of multiplicative approximation for estimate the average in a graph.

Example 2.1. We need to distinguish between a graph with no edge and a graph with just one edge, it is clear that for the first graph the average degree is equal to 0 but for the second one the average degree is more than 0. In order to fix this issue we can add some lower bounds. For example, we can just assume that our input graph is a connected graph or it is a none-empty graph.

Example 2.2. We also need to distinguish between a graph which is a cycle on n nodes with average degree equal to 2 and another graph which is union of a cycle on $n - c\sqrt{n}$ and a clique on $c\sqrt{n}$ where c is a constant greater than 2 with average degree equal to $2 + c^2 - \frac{2c}{\sqrt{n}}$.

In order to fix the second issue we prove the following theorem.

Theorem 1. *Every $c-$approximation algorithm for average degree in a graph $G(V, E)$ with n vertices makes $\Omega(\sqrt{n})$ queries, for every constant c.*

Consider an algorithm A with q queries, we can compute the probability of catching a vertex from the clique part of the second graph is like the following:

$$Pr[A\ catches\ a\ vertex\ from\ the\ clique] \leq q.\frac{c}{\sqrt{n}} \leq c^*$$

Which c^* is equal to some small constant like $\frac{1}{6}$. So if $q \leq \frac{\sqrt{n}}{6c}$ it fails with probability bigger than $\frac{1}{3}$ on at least one of two distributions.

3 Algorithm for Finding the Average Degree

In this chapter we are going to discuss an algorithm for approximating the average degree in a graph. After that we will explain sum important ideas for analysing this algorithm.
First of all, we divide vertices of our graph into $t = \Theta(\frac{1}{\epsilon} \log n)$ buckets B_i for all $1 \leq i \leq k$. Bucket B_i contains all vetices v such that $(1 + \beta)^{i-1} \leq d(v) \leq (1 + \beta)^i$ where $\beta = \frac{\epsilon}{c}$. If $\bar{d}$ was the average degree of our graph G then we will have $\sum_{i=1}^{t} (1 + \beta)^{i-1}|B_i| \leq \bar{d}.n \leq \sum_{i=1}^{t} (1 + \beta)^i|B_i|$.

Then we do the following steps:

1. Sample a set S of vertices where $|S| = \Theta(\sqrt{n}\ poly(\log n, \frac{1}{\epsilon}))$.
2. Put $S_i = S \cap B_i$.
3. Estimate the fraction of vertices, $\frac{|B_i|}{n}$ in each bucket by ρ_i which is set like the following for $1 \le i \le t$:

$$\rho_i = \begin{cases} \frac{|S_i|}{|S|} & \text{if } |S_i| > \sqrt{\frac{\epsilon}{n}}.\frac{|S|}{c.t} \\ 0 & \text{otherwise} \end{cases}$$

4. Output $\sum_{i=1}^{t} \rho_i (1+\beta)^{i-1}$.

Now, we are going to see some important ideas which are useful for the analysis of this algorithm. The first thing is we don't have over estimation with high probability. We call each bucket B_i heavy if $|B_i| > \frac{\sqrt{\epsilon n}.\sqrt{n}.poly().\epsilon}{c' \log n}$ by using Chernoff Bound ($Pr[\frac{1}{m}\sum_{i=1}^{m} X_i > (1+\gamma)P] \le exp(-\frac{\gamma^2 Pm}{3})$) we can prove that $\rho_i \le (1+\gamma)\frac{|B_i|}{n}$. And we also don't have over estimation for light buckets because we put $\rho_i = 0$ in that case.
Now we should discuss the underestimation which says $(1-\gamma)\frac{|B_i|}{n} \le \rho_i$. Again, by using Chernoff bound we can prove that this will holds with high constant probability for all heavy buckets. We can also claim that the vertices in light buckets do not contribute in the final answer too much.

References

[1] Oded Goldreich and Dana Ron. On estimating the average degree of a graph. *Electronic Colloquium on Computational Complexity (ECCC)*, (013), 2004. URL `http://eccc.hpi-web.de/eccc-reports/2004/TR04-013/index.html`.