CSE341T/CSE549T **10/29/2014**

MIS Continued and High Probability Bounds

Lecture 17

1 MIS Continued

We now prove that each round removes a constant fraction of the edges. For each vertex, we define a set $X(v) = \{u : u \in N(v) \cap d(u) \leq d(v)\}$. Therefore $X(v)$ contains a set of neighbors that have degree smaller than or equal to v. A vertex is good if $|X(v)| \geq d(v)/3$, that is, a lot of v's neighbors have "low" degree. Otherwise a vertex is bad. An edge is good if at least one of its endpoints is good. Otherwise it is bad.

Lemma 1 *Say v is a good vertex with $d(v) > 0$. Then the probability that at least one of its neighbors are marked initially is at least $1 - e^{(-1/6)}$.*

Proof. The probability that at least one of v's neighbors is marked is equal to 1 minus the probability that none of its neighbors are marked. Consider any neighbor u of v. The probability that u is marked is $1/2d(u)$. Therefore, the probability that it is not marked is $1 - 1/2d(u)$. Therefore, the probability that none of the neighbors is marked is

$$\prod_{u \in N(v)} (1 - 1/2d(u)) \leq \prod_{u \in X(v)} (1 - 1/2d(u))$$

For all $u \in X(v)$, $d(u) \leq d(v)$. Therefore, the above quantity is less than or equal to

$$\begin{aligned} (1 - 1/2d(v))^{|X(v)|} &\leq (1 - 1/2d(v))^{|d(v)/3|} \\ &\leq e^{-\frac{d(v)/3}{2d(v)}} \end{aligned}$$

Therefore, the probability that some vertex is marked is at least $1 - e^{-1/6}$.

Lemma 2 *Let M be the set of vertices that got marked and let S be the set of vertices that were included in MIS. Then, given a vertex v that was included in M, the probability that it is also in S is at least $1/2$.*

Proof. If v was included in M, it is only removed if it had a neighbor u which was also in M, and $d(u) > d(v)$. Given a vertex u, the probability that it was marked is $1/2d(u)$. The probability

that some such neighbor was marked is at most

$$\begin{aligned} P &\leq \sum_{u \in N(v), d(u) \geq d(v)} \frac{1}{2d(u)} \\ &\leq \sum_{u \in N(v), d(u) \geq d(v)} \frac{1}{2d(v)} \\ &\leq \sum_{u \in N(v)} \frac{1}{2d(v)} \\ &= 1/2 \end{aligned}$$

Lemma 3 *Let V_g be the set of good vertices and S be the set of vertices included in the MIS. The probability that a good vertex v was in either S or the neighbors of S (that is, it disappeared in that round) is at least $1/2(1 - e^{-1/6})$.*

Proof. A vertex v is in the neighborhood of S if there is another vertex u that is neighbor of v and is in S. Since v is a good vertex, we know that some neighbor of v was marked with probability $1 - e^{-1/6}$. Also, we know that the probability that a vertex is unmarked after being marked is at most $1/2$. Therefore, some neighbor of v remained marked with probability at least $1/2(1-e^{-1/6})$. Therefore, v was removed with at least that probability.

Corollary 4 *A good edge is removed with probability at least $1/2(1 - e^{-1/6})$.*

Lemma 5 *At any time, at least half the edges are good.*

Proof. For each edge (u, v) direct it from u to v if $d(u) \leq d(v)$, breaking ties arbitrarily. In the resulting graph, let $d_o(v)$ and $d_i(v)$ be the out and in-degrees. Let V_g and V_b be the set of good and bad vertices. For each bad vertex, $d_o(v) - d_i(v) \geq d(v)/3$. Let E_{bb}, E_{bg}, E_{gb} and E_{gg} be the edges from V_b to V_b, V_b to V_g and so on. The sum of the degrees of bad vertices is

$$\begin{aligned} 2\,|E_{bb}| + |E_{bg}| + |E_{gb}| &= \sum_{v \in V_b} d(v) \\ &\leq 3 \sum_{v \in V_b} (d_o(v) - d_i(v)) \\ &= 3 \sum_{v \in V_g} (d_i(v) - d_o(v)) \\ &= 3((|E_{bg}| + |E_{gg}|) - (|E_{gb}| + |E_{gg}|)) \\ &= 3(|E_{bg}| - |E_{gb}|) \end{aligned}$$

Therefore, $2|E_{bb}| \leq 2|E_{bg}| - 4|E_{gb}|$. Therefore $|E_{bb}| \leq |E_{bg}|$. Other than the edges that go from bad to bad vertices, all other edges are good edges. Therefore, at least half the edges are good.

Theorem 6 *The expected number of edges removed per round is at least* $1/4(1 - e^{-1/6})|E|$.

2 High Probability Bounds

So far, we have generally proved expectation bounds — for example, the running time of Quicksort is $O(n \lg n)$ and the number of rounds in the MST calculation is $O(\lg n)$. However, expectation bounds don't tell us about the probability of failure — that is, apart from the weak results you can get using Markov Inequality, you can not say how frequently your algorithm may run slower than the expectation bounds indicate.

Today, we will think about a stronger way of analyzing randomized algorithms — ***high probability bounds***. We say that an event E takes place ***with high probability*** if

$$\Pr\{E\} \geq 1 - \frac{1}{n^c}$$

for some constant $c \geq 1$. Here n is the instance size. Typically, in bounding the cost of an algorithm, we're interested in showing that $X_n < A$ with high probability. In other words, we want to say

$$\Pr\{X_n < A\} \geq 1 - \frac{1}{n^k}$$

or equivalently,

$$\Pr\{X_n \geq A\} < \frac{1}{n^k}$$

This is saying that the probability of a "bad" event (the random variable takes on a value greater than some value A) is less than $1/n^k$ for some constant k. This probability is very low. We don't normally say something happens "with low probability," because we are interested in the opposite "good" case. For example, for us the X_n will be some cost measure (in our case, the number of rounds), and we want to say that with high probability the cost will be less than some A, i.e., that the probability that it is greater is small.

An important inequality to know for calculating many high probability bounds is ***Chernoff bound***[1].

[1] In the literature, Chernoff bounds are more or less a family of inequalities — a generic term for various bounds that deal with the idea that the aggregate of many independent random variables concentrates around the mean. There are different versions of Chernoff bound; we will see one of them, but the idea is similar; some version gives a tighter bound than others.

Theorem 7 (Chernoff Bound) *Given a set of independent random variables $X_1, \ldots, X_n$, each taking a real value between 0 and 1, with $X = \sum_{i=1}^{n} X_i$ and $\mathrm{E}[X] = \mu$, then for any $\lambda > 0$*

$$\Pr\{X > (1+\lambda)\mu\} \quad < \quad \exp\left(\frac{-\mu\lambda^2}{2+\lambda}\right).$$

First, let's try to understand what the bound says in words. The factor λ specifies a distance away from the mean. So, for example, by setting $\lambda = .2$, we are asking what the probability is that the actual result is more than 20% larger than the expected result. As we expected, the farther we are away from the mean, the less likely our value will be — this probability, in fact, decays exponentially fast in (roughly) λ^2.

Second, independence is key in getting this sharp bound. By comparison, Markov's inequality which doesn't require independence can only give $\Pr\{X > (1+\lambda)\mu\} < \frac{1}{1+\lambda}$, which is a much weaker bound. As weak as Markov's bound is, it is in certain cases the best we can make of if we can't say anything about independence. Indeed, independence is key here: it makes it unlikely that the random variables will "collude" to pull the mean one way or the other.

Now lets try to apply Chernoff bounds to the minimum spanning tree problem. Let R_i be the random variable that represents the number of nodes remaining in the graph after round i. We argued in the previous lecture that $E[R_i] \geq 3n/4$. Therefore, using Markov's Inequality, we can argue that $\Pr\{R_i > 7n/8\} \leq 6/7$.

We now say that a round is successful if at most $7n/8$ nodes remain after it finishes and is unsuccessful otherwise. We know that the total number of successful rounds is at most $\log_8 n$. Therefore, if we define an indicator random variable X_i which is 1 if around is unsuccessful, we get $E[X_i] \leq 6/7$. Say X is a random variable representing the number of unsuccessful rounds — therefore, $X = X_1 + X_2 \ldots$. If we had $\log_8 n$ successful rounds, $E[X] \leq 6 \log_8 n$. Also, all X_i's are independent. Therefore, using Chernoff Bounds, we have

$$\begin{aligned}
\Pr\{X > (1+\lambda)6\log_8 n\} &\leq \exp\left(\frac{-(6\log_8 n)\lambda^2}{2+\lambda}\right) \\
&\leq \exp\left(\frac{-(6\ln n/\ln 8)\lambda^2}{2+\lambda}\right) \\
\Pr\{X > 18\log_8 n\} &\leq n^{-3}
\end{aligned}$$

If we set $\lambda = 2$. As we increase λ, we can make the probability smaller and smaller. Therefore, the probability of having more than $\omega(\lg n)$ unsuccessful rounds (and therefore, $\omega(\lg n)$ total rounds) is very small and gets smaller as n gets larger.

Another reason to use high probability bounds is that they are composable. Say you had the following code:

FOO()

1 **spawn** BAR-1()
2 **spawn** BAR-2()
3 **spawn** BAR-3()
4 ...
5 **spawn** BAR-n()
6 **sync**

To calculate the span of this computation, we need to compose the costs across the three functions by taking the max. Unfortunately, knowing the expectation of contracting each component is not always good enough when composing costs if we are computing the span. In analyzing work, we just add everything up. Linearity of expectations turns out to be very useful for this purpose since if we know the expected work for each component, we can just add them to get the overall expected work. When analyzing span over a parallel computation, however, we take the maximum of the span of the components. Unfortunately the following is (in general) not true: $\max(\mathrm{E}[X], \mathrm{E}[Y]) = \mathrm{E}[\max(X, Y)]$. We therefore cannot simply take the maximum of expectations across components to get the overall expected span.

As an example consider two parallel components each which have span either 1 or 5 each with $.5$ probability. The expected span for each of the components is therefore $(.5 \times 1 + .5 \times 5 = 3$. Now, in combination there are four possible configurations $((1,1),(1,5),(5,1),(5,5))$. Since we take the max of the two components the overall span for the four are $1, 5, 5, 5$. The expected span is the average of these: $.25 \times (1 + 5 + 5 + 5) = 4$. However if we simply took the maximum of the expected span of each component we would have $\max(3, 3) = 3$, which is wrong. The difference can be much greater than this example.

We can deal with this issue by using high-probability bounds instead. In particular, if we have the high-probability bounds, we can use the union bound. The ***union bound*** (Boole's inequality) states that for a set of events $A_1, A_2, \ldots$, that

$$\Pr\left\{\bigcup_i A_i\right\} \leq \sum_i \Pr\{A_i\}.$$

If the events do not overlap (their sets of outcomes do not intersect) then the equation is an equality since we just add up the probabilities of every outcome. If they do overlap then the probability of the union is strictly less than the sum of the probabilities since some outcomes are shared.

Now say that X_i is the random variable that represents the span of BAR-i() and say that we had a high probability bounds that said that $\Pr\{X_i \geq k \lg n\} \leq 1/n^c$. We can use the union bound to say that $\Pr\{\bigcup_i (X_i \geq k \lg n)\} \leq 1/n^{c-1}$, which also implies that the probability that the span of FOO is larger than $k \lg n$ is smaller than $1/n^{c-1}$.