15-750 — Graduate Algorithms — Spring 2008

Miller and Sinop and Wu

Assignment 6 Due date: Friday May 2, 2008

Some Reminders:

- Read the Policies section on the course web site before you start working on this assignment. Collaboration **is** permitted for this assignment.
- You should refrain from using outside sources when solving these problems. For each problem, state whether you have seen it before. If you have questions, contact the course staff.
- We prefer that you type up your solutions (preferably using LaTeX). You may neatly handwrite your solutions, but if we have trouble reading them you will be required to type up future solutions.

Thanks for Bin Fan and Bin Fu for the solutions.

1 Maximum Independent Set in Trees

SOLUTION: (This is a different solution from Gary given. It directly calculate the size of the maximum independent set.)

First, consider storing a vector (a, b) at each node v where a is the number of max independent set not containing v of the tree rooted at v, and b is the number of the max independent set of the same tree, possibly containing v.

The RAKE operation is as follows. Let $(c_1, d_1), \ldots, (c_k, d_k)$ be the vectors stored at the children of v.

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{k} d_i \\ \max\{1 + \sum_{i=1}^{k} c_i, \sum_{i=1}^{k} d_i\} \end{bmatrix}$$

The value of a is just the sum of the weights of the max independent sets at each child. The value of b is the max of a and w_v plus the sum of the weights of the max independent sets at each child that don't contain that child.

If we have evaluated all but one child of v, we use the COMPRESS operation. Let (x, y) denote the vector to be computed at the child that has not yet been evaluated. Let (c, d) be the sum of the vectors stored at the other children. We can now write the compress operation as a matrix operation over the semiring with operations $(max, +)$ replacing the usual operations of $(+, *)$ respectively.

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 0 & d \\ 1 + c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} d + y \\ \max\{1 + c + x, d + y\} \end{bmatrix}$$

It is not hard to show that this is indeed a semiring. Now, we know we can multiply these matrices to get new matrices. This corresponds to collapsing a chain of nodes with exactly one unevaluated child. This suffices to specify the usual parallel tree contraction algorithm. The result to return is b for the root of the tree. The runtime is $O(\log n)$ as with the usual parallel tree contraction.

2 Maximal Independent Set Version II

[20 points]

SOLUTION:

2.1

Since numbers are generated from $[0, n^4]$, the probability of collision would be $\leq \frac{n^2}{n^4} \leq \frac{1}{n^2}$. Let's assume there is no two vertex get the same value. So the probability is $\frac{1}{d(v)+1}$.

For the hint:

Pr(u$\in I, v \in I)$
$= Pr(Value(u) > Value(v), u \in I, v \in I) + Pr(Value(u) > Value(v), u \in I, v \in I)$
$= \frac{1}{\alpha+\beta-\gamma+2}\frac{1}{\alpha+1} + \frac{1}{\alpha+\beta-\gamma+2}\frac{1}{\alpha+1} = \frac{\alpha+\beta+2}{(\alpha+\beta-\gamma+2)(\alpha+1)(\beta+1)}$
Here, we notice that

$Pr(Value(v) > Value(u), u \in I, v \in I)$
$= Pr($u is the smallest among u,v's neighbours , and v is the smallest among its neighbours)

2.2

Assume that u is a random good vertex, so according to the definition of good vertex, $\sum_{v=N(u)} \frac{1}{2d(v)} \geq \frac{1}{6}$. Then there are two different situations:

1. If $\exists v \in N(u)$ and $d(v) \leq 14$. According to the above conclusion, we see that $P(N(u) \in I) \geq P(v \in I) \approx \frac{1}{15}$.

2. If 1 is not satisfied, then assume the neighbors of u are $v_1, v_2, ..., v_\alpha, d(v_i) \geq 15$. Because u is a good vertex, we also know that: $\sum_{i=1}^{\alpha} \frac{1}{2d(v_i)} \geq \frac{1}{6}$. And similar to the classnote, since $d(v_i) \geq 15$ we can find a vertex set M s.t. :

$$\frac{1}{5} = \frac{1}{6} + \frac{1}{30} \geq \sum_{v_i \in M} \frac{1}{2d(v_i)} \geq \frac{1}{6}$$

What we want to calculate is $P(\cup_{i=1}^{\alpha} P(v_i \in I))$, and the following derivation uses the given hint, where $\gamma \leq Min(d(v_i), d(v_j))$ is the number of common vertex between v_i and v_j:

$$P(\cup_{i=1}^{\alpha} v_i \in I)$$

$$\geq P(\cup_{v_i \in M} v_i \in I)$$

$$\geq \sum_{v_i \in M} P(v_i \in I) - \sum_{v_i \in M, v_j \in M, i \neq j} P(v_i \in I \cap v_j \in I)$$

$$\approx \sum_{v_i \in M} \frac{1}{1+d(v_i)} - \sum_{v_i \in M, v_j \in M, i \neq j} \frac{d(v_i)+d(v_j)+2}{(1+d(v_i))(1+d(v_j))(d(v_i)+d(v_j)-\gamma+2)}$$

$$\geq \sum_{v_i \in M} \frac{1}{1+d(v_i)} - \sum_{v_i \in M, v_j \in M, i \neq j} \frac{2}{(1+d(v_i))(1+d(v_j))}$$

$$= \sum_{v_i \in M} \frac{1}{1+d(v_i)}(1 - \sum_{v_j \in M, i \neq j} \frac{2}{1+d(v_j)})$$

$$\geq \sum_{v_i \in M} \frac{1}{1+d(v_i)}(1 - \sum_{v_j \in M} \frac{2}{1+d(v_j)})$$

$$\geq \frac{1}{3}\frac{15}{16}(1 - \frac{2}{5} \times 2)$$

$$= \frac{1}{16}$$

So finally combing 1 and 2, we get to the conclusion that:

$P(N(u) \in I) \geq \frac{1}{16}$ when u is a good vertex.

2.3

With the same operation as the classnote doing Ruby I MIS, next we can show that good edge is at least 1/2 of all edges. And then finally we see that the expected decrease in the number of edges each round is at least $E/32$.

3 Separators for Random Graphs

[20 points]

SOLUTION:

Assume ξ is a Bernoulli random variable with two values 0 and 1 and $E\xi = c/n$.

Let random variable $X_{i,j}$ denotes if there is an edge (V_i, V_j). All $X_{i,j} = \xi$ and i.i.d.

Pick $A \subseteq V$ where $n/3 \leq |A| \leq 2n/3$. Consider the number of edges connecting A and $V \setminus A$, denoted by S, we have:

$$S_A = \sum_{V_i \in A} \sum_{V_j \in V \setminus A} X_{i,j} = |A|(n - |A|)\xi$$

Obviously if $S_A \leq kn$ there is an edge separator with vertex partition A and $V \setminus A$. This probability is:

$$\begin{aligned} P\{S_A \leq kn\} &\leq P\{S_A \leq kn, |A| = n/3\} \\ &\leq e^{-\frac{2}{9}(1-\frac{9k}{2c})^2 cn} \end{aligned}$$

If there is en edge separator, then there is at least one balanced partition A, such that $S_A \leq kn$:

$$\begin{aligned} P\{\exists \text{ edge separator}\} &= P\{\exists A, S_A \leq kn\} \\ &= P\Big\{ \bigcup_{\forall A \subseteq V, n/2 \leq |A| \leq 2n/3} \{S_A \leq kn\} \Big\} \\ &\leq \sum_{\forall A \subseteq V, n/2 \leq |A| \leq 2n/3} P\Big\{ S_A \leq kn \Big\} \\ &\leq 2^n e^{-\frac{2}{9}(1-\frac{9k}{2c})^2 cn} \\ &= e^{(\ln 2 - \frac{2}{9}(1-\frac{9k}{2c})^2 c)n} \end{aligned}$$

When $n \to \infty$, choose k to make $\ln 2 - \frac{2}{9}(1 - \frac{9k}{2c})^2 c < 0$, we get $P\{\exists \text{ edge separator}\} \to 0$. So with high probability, $G(n, c)$ has no edge kn-separator.

4 An NP-Completeness Problem

SOLUTION: First, given a simple cycle in G, we can determine whether the sum of its edge weights is zero in polynomial time. Thus **Zero-Weight-Cycle** $\in$ **NP**.

Then we reduce the **Subset Sum Problem** to this problem. The subset sum problem is: given a set of integers, determine whether the sum of some non-empty subset equal exactly zero. So consider a set of integers $S = \{a_1, \ldots, a_n\}$, we construct a weighted directed graph G with $2n$ vertices, such that every element a_i corresponds to two vertices v_i and u_i. For each v_i, add an edge from v_i to u_i with weight a_i and add edges from every vertex u_j to it with weight 0. For each u_i, add edges from this vertex u_i to every other v_j with weight 0. If we find a zero-weight-cycle in G, then all the weights from v_i to u_i along the cycle must be zero. On the other hand, if we get a subset $S' \subseteq S$ which sums to zero, we construct a cycle by picking all edges (v_i, u_i) corresponds to the element in S' and connect those edges by those zero weight edges and finally obtain a zero weight cycle. Thus this problem is at least as hard as subset sum problem. Since the subset problem is NP-complete, we have **Zero-Weight-Cycle** $\in$ **NPC**.