CMSC 858S: Randomized Algorithms

Fall 2001

Project

Due date: Beginning of class on December 7, 2001

General instructions: Please work in groups of two, with at most one group of three. Please type up your solution and submit your code. As always, partial credit will be given where appropriate. So if you are unable to solve a problem but can do so by making some assumptions, please clearly state these assumptions and proceed.

The basic problem we consider is the following. We have a set V of n elements, and m distinct subsets $S_1, S_2, \ldots, S_m$ of V, each having cardinality t. Our goal is to choose a subset W of V with "many" elements, subject to the constraint that no S_i (for $i = 1, 2, \ldots, m$) be a subset of W.

Consider the following algorithm $\mathcal{A}$ for this problem. Let V be the set $\{1, 2, \ldots, n\}$. Independently for each $i \in V$, choose a number X_i uniformly at random from the set $\{1, 2, \ldots, n^3\}$. Now define a set W as follows: for each $i \in V$, $i \in W$ iff there is no set S_j such that: (i) $i \in S_j$, and (ii) for all $k \in S_j$, $X_i \geq X_k$.

1. Show that $\mathcal{A}$ always produces a feasible solution to our problem. **(1 point)**

2. Suppose $i \in V$ lies in a_i of the sets $S_1, S_2, \ldots, S_m$. Show that the expected size of the set W produced by $\mathcal{A}$ is at least

$$(1/n^3) \cdot \sum_{i=1}^{n} \sum_{\ell=1}^{n^3} \left(1 - (\ell/n^3)^{t-1}\right)^{a_i}. \tag{1}$$

Hint: Denote by B_i the event that $i \in W$, and first try to get a lower bound on the value $\mathbf{E}[B_i \mid (X_i = \ell)]$ for an arbitrary ℓ. Can you apply some correlation inequality that we studied, in order to get this lower bound? Next, once you have a lower bound on $\mathbf{E}[B_i \mid (X_i = \ell)]$, show how to use it to get a lower bound on $\mathbf{E}[|W|]$. **(4 points)**

3. Find out how to generate random numbers in C or C++ in an operating system that you are familiar with. Implement a C or C++ program which will: (a) take the sets V and $S_1, S_2, \ldots, S_m$ as input, (b) run the above randomized algorithm independently 100 times, and (c) output the "best" (largest) set W that is constructed by these 100 runs. Do this for 5 data sets of your choice with $n = 50$, $500 \leq m \leq 600$, and $6 \leq t \leq 8$. Please submit your code (at most 200 lines please!), the results of these 5 datasets, and how they compare with the bound shown by (1) above. **(5 points)**