

# CS 361A - Advanced Data Structures and Algorithms

Autumn Quarter, 2005

Homework #4 (Due Date: 12/9/05)

- (25 points) In defining a random leveling for a skip list, we sampled the elements from  $L_i$  with probability  $1/2$  to determine the next level  $L_{i+1}$ . Consider instead the skip list obtained by performing the sampling with probability  $p$  (at each level), where  $0 < p < 1$ .
  - Determine the expectation of the number of levels  $r$ , and prove a high probability bound on the value of  $r$ .
  - Determine as precisely as you can the expected cost of the *find* operation in this skip list.
  - Discuss the relation between the choice of the value  $p$  and the performance of the skip list in practice.
- (35 points) We would to maintain a sketch for the count vector  $\{m_1, m_2, \dots, m_n\}$ . The purpose of maintaining the sketch is to estimate the fourth frequency moment  $F_4 = \sum_i m_i^4$ . We are given a family of 8-wise independent hash functions defined over the index domain  $[n] = \{1, 2, \dots, n\}$  corresponding to the count vector. However, unlike the technique from Lecture 16 where the range of the hash functions was  $\{-1, 1\}$ , the range of the hash functions in our case is the set of complex values  $\{1, -1, j, -j\}$ , where  $j$  is the fourth root of unity, i.e.,  $j^4 = 1$ . A hash function chosen uniformly from this family maps any value  $i$  to the four values in the range with equal probability ( $1/4$  each); also, as mentioned earlier, the hash functions are 8-wise independent. Such hash function families exist and a hash function from this family can be represented using only  $O(\log n)$  space.

Consider the random variable  $X = \sum_i m_i Z_i$  where  $Z_i = h(i)$  is the value of the hash function  $h$  evaluated at  $i$ . We can maintain the random variable  $X$  efficiently in a data stream model, as in the technique from Lecture 16. This requires 2 words of size  $\log m$  bits each, where  $m$  is the length of the stream, one each for the real and complex part.

- Prove that  $E[Y = X^4] = F_4$ .
- Also prove,  $Var(Y) = \binom{8}{4} - 2)(\sum_{i < j} m_i^4 m_j^4)$ .

The last two assertions *appear* to hold the promise that we can estimate the fourth moment ( $F_4$ ) using the *median of averages* trick as before. Thus, one may be persuaded that we can estimate  $F_4$  using small space (polylogarithmic in  $n$ ) in the data stream model. However, we have seen that there is a space lower bound of  $\Omega(n^{1/4})$  for estimating  $F_4$  in the data stream model. Find the flaw in the preceding argument and explain why we cannot circumvent the lower bound.

3. (35 points) Recall the *update* model for data stream introduced at the beginning of Lecture 16. We are given a stream of data elements which are pairs of the form  $(i, a_{ij})$ , where  $i \in [n] = \{1, 2, \dots, n\}$  and  $j > 0$  is just a unique index that differentiates the different updates corresponding to the same  $i$ . Assume that all  $a_{ij}$ 's are positive integers. At any instant in time, these pairs define an implicit vector  $V_{sum}$  whose  $i$ th coordinate is given by  $V_{sum}(i) = \sum_j a_{ij}$ . Effectively, the  $i$ th coordinate is obtained by taking the sum of the values  $a_{ij}$  from the pairs  $(i, a_{ij})$  corresponding to  $i$ . Instead of taking the sum of  $a_{ij}$ 's corresponding to a specific  $i$ , if we took the max value amongst them we would obtain the vector  $V_{max}$  whose  $i$ th coordinate is given by  $V_{max}(i) = \max_j a_{ij}$ .

Note that each coordinate of  $V_{max}$  is a positive integer, since  $a_{ij}$ 's are positive integers. We would like to compute the  $l_1$  norm of this vector in the data stream model, namely  $l_1(V_{max}) = \sum_i |V_{max}(i)| = \sum_i V_{max}(i)$ .

Suppose that we are given a data structure  $D$  for computing the count of distinct elements seen so far in a data stream, where each element in the data stream is drawn from the universe  $[n] = \{1, 2, \dots, n\}$ . Suppose further that  $R$  is the max value that any  $a_{ij}$  can take. Show how to compute  $l_1(V_{max})$  to within a multiplicative ratio of 2, using  $\log R$  instances of the data structure  $D$ . The estimate  $f$  produced by the algorithm should satisfy  $l_1(V_{max})/2 \leq f \leq 2l_1(V_{max})$ .

4. (35 points) Suppose you have two data streams  $X = \{x_1, x_2, \dots\}$  and  $Y = \{y_1, y_2, \dots\}$ , where  $x_i, y_j \in U$  are drawn from some universe  $U$ . Let  $X'$  and  $Y'$  denote the set of distinct elements that appear in the streams  $X$  and  $Y$  respectively. We define the similarity between the two streams as  $sim(X, Y) = |X' \cap Y'| / |X' \cup Y'|$ .

Assuming  $sim(X, Y) > c$ , where  $c$  is a constant fraction. Given user-specified parameters  $\delta > 0$  and  $\epsilon > 0$ , design a randomized stream algorithm to estimate  $sim(X, Y)$  to within an additive error of  $\epsilon$  with probability at least  $1 - \delta$ . Your algorithm should be as efficient in terms on memory usage as possible – be sure to analyze the memory requirement of your algorithm.