Sublinear Time and Space Algorithms 2018B – Problem Set 1

Robert Krauthgamer

Due: April 15, 2018

General instructions: Please keep your answers short and easy to read. You can use results, calculations or notation seen in class without repeating them, unless asked explicitly to redo them.

1. Design an algorithm that samples s items without replacement from an input stream $\sigma = (\sigma_1, \ldots, \sigma_m)$. The algorithm's memory requirement should be $O(s)$ words (s is a parameter known in advance). Prove that the algorithm's output has the correct distribution.

   Hint: The goal is essentially to sample s distinct indices ($i_1 < \cdots < i_s$) uniformly at random. In contrast, executing the Reservoir Sampling algorithm s times in parallel gives k samples *with* replacement, i.e., the same $i \in [m]$ could be reported more than once.

2. Recall that Algorithm Bottom-k seen in class reports $X := k/z_k$ as an estimate for the number of distinct elements $d^* := \|x\|_0$ (where $x \in \mathbb{R}^n$ denotes the frequency vector of the input stream). Prove that for suitable $k = O(1/\varepsilon^2)$,

   $$\Pr[X > (1+\varepsilon)d^*] \leq 0.05,$$
   $$\Pr[X < (1-\varepsilon)d^*] \leq 0.05.$$

   Hint: Introduce Y to count how many hash values are below the threshold $\frac{k}{(1+\varepsilon)d^*}$, and analyze its expectation and variance.