Algorithm Design 21/22

Hands On 7 - Deterministic Data Streaming

Federico Ramacciotti

1   Problem

Consider a stream of n items, where items can appear more than once. The problem is to find the most frequently appearing item in the stream (where ties are broken arbitrarily if more than one item satisfies the latter). For any fixed integer $k \geq 1$, suppose that only k items and their counters can be stored, one item per memory word; namely, only $O(k)$ memory words of space are allowed.

Show that the problem cannot be solved deterministically under the following rules: any algorithm A can only use these $O(k)$ memory words, and read the next item of the stream (one item at a time). You, the adversary, have access to all the stream, and the content of these $O(k)$ memory words: you cannot change these words and the past items, namely, the items already read, but you can change the future, namely, the next item to be read. Since any algorithm must be correct for any input, you can use any amount of streams and as many distinct items as you want.

Hints:

1. This is "classical" adversarial argument based on the fact that any deterministic algorithm A using $O(k)$ memory words gives the wrong answer for a suitable stream chosen by the adversary.

2. The stream to choose as an adversary is taken from a candidate set sufficiently large: given $O(k)$ memory words, let $f(k)$ denote the maximum number of possible situations that algorithm A can discriminate. Create a set of C candidate streams, where $C > f(k)$ and $S2$ cannot distinguish, by the pigeon principle.

2   Solution

Given an algorithm A and a function $Freq$ that exactly finds the most frequent item in a string, the goal is to find two streams $\tilde{S}_1, \tilde{S}_2$ s.t. $Freq(\tilde{S}_1) = Freq(\tilde{S}_2) \wedge A(\tilde{S}_1) = A(\tilde{S}_2)$.

Let's take the universe U and its subsets $\Sigma_i \subseteq U$ s.t. $|\Sigma_i| = \left\lceil \frac{|U|}{2} \right\rceil$. With $|\Sigma_i| = \frac{|U|}{2} + 1$ the number of possible subsets is

$$\binom{|U|}{\frac{|U|}{2} + 1} \cong 2^{|U|} > f(k)$$

Select from this subsets two subsets Σ_i and Σ_j such that $|\Sigma_i \cap \Sigma_j| \geq 1 \wedge |\Sigma_i \setminus \Sigma_j| \geq 1$, so that they have at least one character in common and at least one character different.

Define two streams S such that

$$\exists \, S_i \in \Sigma_i^*, \; S_j \in \Sigma_j^* \; s.t. \; S_i \neq S_j \wedge S_i \stackrel{A}{=} S_j$$

so that S_i and S_j are indistinguishable for the algorithm A.

Given the streams

$$S_i = s_1^i s_2^i \ldots s_{\frac{|U|}{2}}^i$$

$$S_j = s_1^j s_2^j \ldots s_{\frac{|U|}{2}}^j$$

we choose two characters x, y such that $x \in \Sigma_i, \; x \notin \Sigma_j$ and $y \notin \Sigma_i, \; y \in \Sigma_j$ and we concatenate them to the streams. Thus, $A(S_i x y) = A(S_j x y)$ because $S_i \stackrel{A}{=} S_j$ and, since $Freq(S_i x y) \neq Freq(S_j x y)$, we can conclude that the algorithm fails.