Advanced Algorithms. Home Exam 2019/20

The Problem

This problem connects two main topics of the course: approximation and randomization.

Background. In real-time environments it can be an important skill to solve cognitive tasks quickly, for instance, estimate how many objects of a certain type are present. Imagine that a picture with a cloud of dots is shown to you, and you are supposed to say how many dots you see. (We call them "dots" in order to distinguish them from arbitrary points.) But you do not have the time to count them – how do you proceed? Perhaps you look at different parts of the picture and (subconsciously) estimate the distances between the dots there, as the basis of a global estimate. We will study a simplified one-dimensional toy version of this problem.

Scenario. In the unit interval $[0, 1]$, an unknown huge number n of dots are located at n unknown fixed points. You have a sensor, and you can put it on any point $p \in [0, 1]$. Then the sensor will return the coordinates ℓ and r of the dots closest to p to the left and to the right, respectively. That means: $\ell \leq p \leq r$, and no other dots are in the interval $[\ell, r]$. (If you incidentally choose a position p of a dot, the sensor will return $\ell = p$ and r as specified above.) We call such an action a"probe" and assume that it takes constant time. Moreover, probes are the only way to gather information.

The task is to estimate n by doing and carefully analyzing some probes. But since n is huge, you can afford only a number of probes that is much smaller than n.

Note that absolutely nothing is known about the dots. However, for some physical reasons, no two dots can have a distance smaller than d, where d is some small number that is known in advance. (Clearly, it follows $n < 1/d$, but n could be any number below this bound.)

Submission

Mail your final report to `ptr@chalmers.se` as PDF attachment (no other formats please). Write your name, ID number, and study programme on the title page. The final submission deadline is annoucned on the course web page. Do not wait until the last minute, but submit when you are done. See also the further Instructions below.
Quality is more important than quantity. However, as a rule of thumb, your report should have at least 5 pages of text (with usual font size, spacing, and margins), plus the title page and possible references.

Some Detailed Suggestions

Design an algorithm that specifies which points you probe and what calculations you do in order to output an approximate value close to n. You may think in these directions and address these questions:
Intuitively, the smaller the typical distances $r - \ell$ are, the larger is n. But can you simply use some aggregated information, e.g., the average of the measured distances?
Can any deterministic algorithm succeed? (Note that many dots may be "hidden" in a segment that the algorithm does not investigate ...) Maybe you must select the points p to be probed by some randomized rule?
But you **must not** assume that the dots' positions are random, too (e.g., uniformly distributed or so). Rather, they are arbitrary, which is not the same as random! In particular, different segments may have high or low densities of dots, in arbitrary ways.
Is the assumption of a minimum distance d crucial? What can happen if we drop this assumption?
For your proposed algorithm, can you prove something about the approximation ratio (estimated n versus true n), or about the probabilty of large deviations from n, as a function of the number of probes?
Feel free to extend the work or to add more reflections. (For instance: Can you even achieve a randomized PTAS for the problem? What about generalizations to 2 dimensions? etc. etc.)

Criteria for a Good Report

- Correctness: There are no major factual mistakes. In particular, you avoid invalid calculations with probabilities and random variables. The final "product" does not have to be perfect, but what you write must be sound.
- Depth: You provide some solid, substantial results that are fully worked out, not only some trivial observations or vague heuristic guesses.
- Clarity: Algorithms, as well as proofs of their properties, are well described. One can follow your arguments step by step. (See also the general grading criteria on the course web page.)
- Negative statements (that something is probably not possible to do) are motivated by good reasons.

Instructions

- Do not misunderstand the above suggestions as sub-exercises that have to be solved one by one. You can choose your specific working directions, as long as you stick to the main goal.
- Do not hesitate to discuss, and to send questions or drafts, in order to check whether you are on the right track, and to get early feedback. But take availability times into account; see the course web page.
- You can submit arbitrarily many drafts, at any time. Only the last version submitted before the deadline will count for your grade. In fact, **drafts are strongly encouraged**. The worst strategy would be to start thinking only a few days before the deadline – this would most likely result in a very poor report. The details of the problem may be more tricky than anticipated at first glance.
- Utmost academic honesty is expected. The words about cheating (see the course web page) apply also here. In particular, you must cite all literature you have used, and acknowledge all sources of help, and always describe the contents in your own way.