Institut for Matematik og Datalogi
Syddansk Universitet

May 14, 2019
JFB

Online Algorithms – F19 – Assignment 3

Assignment due Tuesday, May 21, 10:15

This is the third of three sets of problems (assignments) which together with the oral exam in June constitute the exam in DM860. This third set of problems may be solved in groups of up to three.

The assignment is due at 10:15 on Tuesday, May 21. You may write this either in Danish or English. If your assignment is late, it will not be accepted. Turn in the assignment through the SDU Assignment system in Blackboard, and remember to keep your receipt. Turn in one PDF file per group.

Cheating on this assignment is viewed as cheating on an exam. Do not talk with anyone outside of your group (or Joan Boyar) about the assignment, and do not show your solutions to anyone outside your group. If you have questions about the assignment, come to Joan Boyar.

Explain all of your answers.

1. In the following, we are considering the online vertex coloring problem. A graph arrives online, so its vertices arrive one by one, and when a vertex v arrives, the neighbors of v which have already arrived are specified, with the request. When the online algorithm receives the request v, it must assign a color $c \in \{1, 2, \ldots\}$ to v, such that c is different from any color assigned to some neighbor of v. (Thus, when done, for any edge in the graph, its endpoints are assigned different colors.) The goal is to use as few colors as possible.

   The algorithm SimpleGreedy always assigns the lowest possible color (thus the color 1 if none of its neighbors have color 1).

   The algorithm BipartiteGreedy is only applied to bipartite graphs (graphs for which the vertices can be partitioned into two parts, U and V, such that both U and V are independent sets in the graph). For a vertex v, BipartiteGreedy first determines which connected component in the graph seen so far v belongs to. (A connected component is a subgraph for which any two vertices in that subgraph are connected by a path.) Note that this subgraph is bipartite. Give v the lowest numbered color not occurring in the part of that connected component where v does not belong.

(a) Give a sequence for which the graph is a tree, where BIPARTITEGREEDY uses 4 colors.

(b) Give a sequence for which the graph is a tree, where SIMPLEGREEDY and BIPARTITEGREEDY give different colorings.

(c) Prove that the competitive ratio of SIMPLEGREEDY is $\Omega(n)$ on bipartite graphs.

(d) Prove that the competitive ratio of SIMPLEGREEDY is at most $\Delta/2$ on bipartite graphs with maximum degree Δ.

(e) Suppose the graph has n vertices and is bipartite. Give an online algorithm using n bits of advice which is optimal. Explain.

2. For a positive integer k, consider the special case of the Bin Packing problem where each input size for $i \in \{1, 2, \ldots, n\}$ is given as a binary fraction with exactly k bits of precision, so

$$x_i = 0.b_1^i b_2^i \cdots b_k^i,$$

where $b_j^i \in \{0, 1\}$. (You can convert it to a decimal fraction as follows: $x_i = \sum_{j=1}^{k} b_J^i (\frac{1}{2})^j$.) Design an online algorithm with $O(f(k) \log n)$ bits of advice that solves this special case of the bin packing problem optimally. Here $f(k)$ is a function that depends only on k. Thus, for each fixed value of k, your algorithm uses $O(\log n)$ bits of advice.

(a) Give your algorithm (more English than pseudocode is fine) and argue that it achieves optimality.

(b) Analyze the length of advice of your algorithm. What is the function $f(k)$?

3. Show that the CLIQUE problem is AOC Complete. (The input to the CLIQUE problem is vertices of a graph and when a vertex v arrives, the neighbors of v which have already arrived are specified, with the request. The accepted vertices must form a clique, so there must be an edge between every pair of vertices accepted. The goal is accept as many vertices as possible.)

4. Is the following statement true or false? If it is true, prove it. If it is false, give a counterexample, along with a restriction of the problem where it is true:

   Let $\sigma = \langle x_1, x_2, \ldots, x_n \rangle$ be an arbitrary request sequence for the k-server problem, where each x_i belongs to some fixed set of $k+1$ difference possible points. Then $OPT(\sigma) \leq \frac{1}{k} \sum_{i=1}^{n-1} d(x_i, x_{i+1}) + o(OPT(\sigma))$.