Advanced Algorithms — Problem set 5

Whenever you give an algorithm, also argue for its running time and correctness. Always feel free to ask in the forum or send a mail if you're stuck, need a clarification, or suspect a typo.

1. Let us attempt to transfer the greedy strategy for Set Cover to the Knapsack problem: In the following, define the "quality" (or "value density") of a Knapsack item i as the quotient $q_i = p_i/w_i$ of profit over weight. Consider the items to be sorted in decreasing order by quality.

   (a) Show that greedily including items of highest quality (starting from an empty set) can give arbitrarily bad solutions. Formally, show that, for every $\epsilon > 0$, there is an input for which this greedy algorithm outputs a solution of profit at most ϵ times the optimum.

   (b) We modify the greedy algorithm: Let $k \leq n$ be the first index such that items $1, \ldots, k$ (in the item list that is sorted by decreasing quality) have total weight strictly larger than the capacity B. Then output the more profitable set among the following two:

   - the first $k-1$ items, or
   - item k alone (we assume that every item has weight $\leq B$).

   Show that this solution gives at least half of the optimal profit. (Hint: It may conceptually help to visualize every item i as a rectangle of width w_i, height q_i, and thus area p_i. Then imagine the items arranged on a horizontal line, ordered by quality. Also imagine a vertical line at B.)

2. The Maximum Independent Set problem is NP-hard, and assuming $\mathsf{P} \neq \mathsf{NP}$, there is not even a polynomial-time algorithm that outputs an independent set of size $\frac{\text{OPT}}{n^{1/4}}$.

   (a) In this part, we restrict the problem to graphs G of constant maximum degree. That is, there is some constant Δ such that every vertex in G is incident with at most Δ edges. Maximum Independent Set remains NP-hard on such graphs, even for $\Delta = 3$. Give an algorithm that always outputs an independent set of size at least

   i. $\frac{\text{OPT}}{\Delta+1}$ (Hint: greedily choose vertices to include into the independent set, lower-bound the number of vertices chosen, and upper-bound OPT)

   ii. $\frac{\text{OPT}}{\Delta}$ (Hint: same as before, but find a tighter upper bound on OPT—trickier)

   (b) In this part, we restrict the problem to *planar* graphs. Such graphs can be drawn in the plane without crossings, and Maximum Independent Set remains NP-hard on such graphs. A famous theorem establishes that planar graphs are 4-colorable, i.e, their vertices can be colored using 4 colors such that the two endpoints of any edge receive distinct colors. Assuming you get an algorithm that outputs a 4-coloring of a planar graph, show how to construct an algorithm that always outputs an independent set of size at least $\frac{\text{OPT}}{4}$.

   (c) Show that all planar graphs are 6-colorable, and describe a polynomial-time algorithm for finding a 6-coloring in a planar graph. You may use that planar graphs are 5-*degenerate*, which means that every planar graph contains at least one vertex of degree ≤ 5.