Advanced Algorithms - Problem set 3

Whenever you give an algorithm, also argue for its running time and correctness. Always feel free to ask on MS Teams if you're stuck, need a clarification, or suspect a typo.

1. Implement the approximate DNF-SAT counting algorithm (Karp-Luby sampling) from the lecture in a language of your choice. Your code should contain functions for

   (a) uniformly sampling from the sample space of pairs (i, a) where i is a clause index and a is an assignment satisfying clause i,

   (b) checking whether a sample (i, a) is "good", i.e., checking whether i is the minimal index such that a satisfies clause i.

   Run the algorithm on three interesting DNF formulas (with an appropriate number of variables and clauses) and plot the output of the algorithm as the number of samples grows.

2. Solve Exercise 1.1 in Vazirani's book.

3. In this exercise, we *rule out* approximation algorithms for the Traveling Salesperson Problem.[1]

   (a) Assuming $\mathsf{P} \neq \mathsf{NP}$, show that there is no polynomial-time 2-approximation algorithm for TSP, even when the edge-weights are all non-zero. (Hint: Reduce from the Hamiltonian Cycle problem. Given a graph G for this problem, construct a weighted graph G' for TSP such that a 2-approximation algorithm for TSP allows you to infer whether G has a Hamiltonian cycle.)

   (b) Assuming $\mathsf{P} \neq \mathsf{NP}$, show that there is not even a polynomial-time $p(|V|)$-approximation algorithm (for any polynomial p) for TSP.

4. 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 $\frac{\mathrm{OPT}}{n^{1/4}}$.

   (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. $\frac{\mathrm{OPT}}{\Delta+1}$ (greedily put vertices into the independent set, give a lower bound on the number of vertices picked this way, and a trivial upper bound on OPT)

      ii. $\frac{\mathrm{OPT}}{\Delta}$ (same approach as before, but find a tighter upper bound on OPT—this part may be 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 $\frac{\mathrm{OPT}}{4}$.

[1] Recall that TSP asks, given a graph $G = (V, E)$ with edge-weights $w : E \to \mathbb{N}$, to find a Hamiltonian cycle in G of minimum total weight.