Chocolate 3

Steve Vott & Winston (Hanting) Zhang

September 26, 2022

Problem 3. [0]

Chocolate Problem: 2 chocolate bars

Reminder: If you solve a chocolate problem (which you can do in groups of size up to 3), please e-mail David with the solution — do not submit it on Gradescope. Also, feel free to list preferences or dietary restrictions for/against particular types of chocolate.

Exercise 4.31 in the textbook. Notice that Part (b) is really the interesting thing here — Part (a) is basically a slightly harder regular problem.

Proposition 1. *Prove that for every pair of nodes $u, v \in V$, the length of the shortest $u - v$ path in H is at most 3 times the length of the shortest $u - v$ path in G.*

Proof. Denote the weight of a edge $e = (u, v)$ by $w(e) = w(u, v)$. For some subgraph $K \subseteq G$, denote the length of the shortest $u - v$ path in K by $d_K(u, v)$. Let H be the output of our algorithm. Suppose for the sake of contradiction that there exists some $u, v \in V$ such that $d_H(u, v) > 3d_G(u, v)$. Let the $u - v$ path in G be made up of the sequence of vertices $u = v_1, v_2, \ldots, v_k = v$. Since H is connected, for each edge of the path (v_i, v_{i+1}), for $1 \leq i < n$, there is a path in H connecting v_i and v_{i+1}.

We claim that $d_H(v_i, v_{i+1}) \leq 3d_G(v_i, v_{i+1}) = 3w(v_i, v_{i+1})$. The equality holds because the edge (v_i, v_{i+1}) itself is the shortest path between v_i and v_{i+1}. If it weren't, then we could improve $d_G(u, v)$ by taking the shorter path between v_i and v_{i+1}. As a corollary, this implies that $d_G(u, v) = \sum_{i=1}^{n-1} w(v_i, v_{i+1}) = \sum_{i=1}^{n-1} d_G(v_i, v_{i+1})$.

We split into two cases.

1. If $(v_i, v_{i+1}) \in H$, then clearly $d_H(v_i, v_{i+1}) \leq 3d_G(v_i, v_{i+1})$.
2. If $(v_i, v_{i+1}) \notin H$, then consider the step of our algorithm when we are have the (incomplete) graph H' and are considering adding the edge (v_i, v_{i+1}). We deduce that v_i and v_{i+1} must be connected at this point, else we would add (v_i, v_{i+1}) into H'. Because (v_i, v_{i+1}) was not added, we deduce that $d_{H'}(v_i, v_{i+1}) \leq 3w(v_i, v_{i+1})$. Since $H' \subseteq H$, we have $d_H(v_i, v_{i+1}) \leq d_{H'}(v_i, v_{i+1})$ (intuitively, we can always do better when we have more edges). Thus $d_H(v_i, v_{i+1}) \leq d_{H'}(v_i, v_{i+1}) \leq d_G(v_i, v_{i+1})$, as desired.

Thus we have,

$$d_H(u, v) \leq \sum_{i=1}^{n-1} d_H(v_i, v_{i+1}) \leq 3 \sum_{i=1}^{n-1} d_G(v_i, v_{i+1}) = 3d_G(u, v),$$

and the proof is complete. □

Proposition 2. *Despite its ability to approximately preserve shortest-path distances, the subgraph H produced by the algorithm cannot be too dense. Let $f(n)$ denote the maximum number of edges that can possibly be produced as the output of this algorithm, over all n-node input graphs with edge lengths. Prove that*

$$\lim_{n \to \infty} \frac{f(n)}{n^2} = 0.$$

Proof. We have no idea. □