Principles of Distributed Computing 05/15, 2019

Exercise 11

Lecturer: Mohsen Ghaffari

Cut Edges

Exercise 1: Given a graph $G = (V, E)$, an edge $e \in E$ is called a *cut edge* if its removal disconnects G. In this exercise, we devise an $O(D)$ round distributed algorithm (with $O(\log n)$-bit messages) that identifies all cut edges, where D denotes the diameter of G.

Let T be a BFS tree of G. For an edge $e \in T$ and an edge $e' \notin T$, we say e' covers e if and only if the unique cycle in $T \cup \{e'\}$ contains e. That is, in $T \cup \{e'\}$, there is a way to go from one endpoint of e to its other endpoint, without passing through e. In fact, any nontree edge $e' = \{u, v\} \notin T$ covers all tree edges e which are on the path from u to w (or from v to w), where w is the lowest common ancestor of u and v.

(a) Argue that any cut edge in G must be a tree edge $e \in T$ for which there is not nontree edge $e' \notin T$ that covers e.

(b) Devise an algorithm that in $O(D)$ rounds, makes each node v know all of its at most D ancestors.

(c) Use the above item to argue that in $O(D)$ rounds, we can make the endpoints u, v of any nontree edge $e' = \{u, v\}$ know their lowest common ancestor w, at the same time for all nontree edges.

(d) Use the above items to complete an $O(D)$ round algorithm for identifying all cut edges.

Orienting Long Trees

Exercise 2: Consider an arbitrary tree T in a graph $G = (V, E)$. Suppose that G has diameter D. However, T can have arbitrarily large diameter, up to $n - 1$. Suppose that we are given a root node $r \in V$. We devise a randomized algorithm that in $O(D + \sqrt{n} \log n)$ rounds, orients all edges of T toward the root r, with high probability. First, mark each edge of T with probability $\frac{1}{\sqrt{n}}$.

(a) Prove that, with high probability, at most $O(\sqrt{n} \log n)$ edges are marked. If you cannot prove the statement with high probability, at least argue that the expected number of marked edges is at most $\sqrt{n}$.

(b) Prove that, with high probability, each of the connected components of T remaining after the removal of marked edges has diameter at most $O(\sqrt{n} \log n)$.

(c) Devise an $O(\sqrt{n} \log n)$ round algorithm that identifies the connected components remaining after the removal of marked edges.

(d) Using the connected components identified above, devise an $O(D + \sqrt{n} \log n)$ round algorithm that orients all marked edges toward the root. Think about gathering marked edges and which components they connect to a central node.

1. Assume that marked edges are oriented toward the root. Devise an algorithm that, in $O(D + \sqrt{n} \log n)$ extra rounds, orients the edges inside the components, all at the same time, toward the root.