Principles of Distributed Computing	03/27, 2019
Exercise 6	
Lecturer: Mohsen Ghaffari	

Network Decompositions

Exercise 1: Explain how given a $(\mathcal{C}, \mathcal{D})$ network decomposition of graph G, we can deterministically compute a $(\Delta + 1)$-coloring of the graph in $O(\mathcal{C}\mathcal{D})$ rounds. Here, Δ denotes an upper bound on the maximum degree of the graph, and is given to the algorithm as an input.

Exercise 2: In this exercise, we prove that every n-node graph G has an $(\mathcal{C}, \mathcal{D})$ (strong-diameter) network decomposition for $\mathcal{C} = O(\log n)$ and $\mathcal{D} = O(\log n)$. The process that we see that be viewed as a simple and efficient sequential algorithm for computing such a network decomposition.

We determine the blocks G_1, G_2, ..., $G_{\mathcal{C}}$ of network decomposition one by one, in C phases. Consider phase i and the graph $G \setminus (\cup_{j=1}^{i-1} G_j)$ remaining after the first $i-1$ phases which defined the first i blocks $G_1, \ldots, G_{i-1}$. To define the next block, we repeatedly perform a ball carving starting from arbitrary nodes, until all nodes of $G \setminus (\cup_{j=1}^{i-1} G_j)$ are removed. This ball carving process works as follows: consider an arbitrary node $v \in G \setminus (\cup_{j=1}^{i} G_j)$ and consider gradually growing a ball around v, hop by hop. In the k^{th} step, the ball $B_k(v)$ is simply the set all nodes within distance k of v in the remaining graph. In the very first step that the ball does not grow by more than a 2 factor — i.e., smallest value of k for which $|B_{k+1}(v)|/|B_k(v)| \leq 2$ — we stop the ball growing. Then, we carve out the inside of this ball — i.e., all nodes in $B_k(v)$ — and define them to be a cluster of G_i. Hence, these nodes are added to G_i. Moreover, we remove all boundary nodes of this ball —i.e., those of $B_{k+1}(v) \setminus B_k(v)$—and from the graph considered for the rest of this phase. These nodes will never be put in G_i. We will bring them back in the next phases, so that they get clustered in the future phases. Then, we repeat a similar ball carving starting at an arbitrary other node v' in the remaining graph. We continue a similar ball carving until all nodes are removed. This finishes the description of phase i. Once no node remains in this graph, we move to the next phase. The algorithm terminates once all nodes have been clustered.

Prove the following properties:

1. Each cluster defined in the above process has diameter at most $O(\log n)$. In particular, for each ball that we carve, the related radius k is at most $O(\log n)$.
2. In each phase i, the number of nodes that we cluster —and thus put in G_i — is at least 1/2 of the nodes of $G \setminus (\cup_{j=1}^{i} G_j)$.
3. Conclude that the process terminates in at most $O(\log n)$ phases, which means that the network decomposition has at most $O(\log n)$ blocks.

Exercise 3 (*optional*): Develop a deterministic distributed algorithm with round complexity $2^{O(\sqrt{\log n \cdot \log \log n})}$ for computing an $(\mathcal{C}, \mathcal{D})$ (strong-diameter) network decomposition in any n-node network, such that $\mathcal{C} = O(\log n)$ and $\mathcal{D} = O(\log n)$.