SS 2007Prof. R. Wattenhofer / Prof. P. Widmayer / Prof. S. Suri / D. Bilò/ T. Locher / Y. A. Oswald

Principles of Distributed Computing
Exercise 12: Sample Solution

We provide a solution for Task 2, which includes a solution for Task 1 as a special case. The construction is the following

1) Given G, compute a rooted spanning tree T.
2) Set $\alpha = (\epsilon n)/2k$.
3) For a node v, let $C(v)$ be the children of v in T.
4) The algorithm processes nodes from the bottom on, and computes a weight $W(v)$ for each node as follows.

$$W'(v) = 1 + \sum_{u \in C(v)} W(u).$$

5) If $W'(v) >= \alpha$, then add v to detector set; and put $W(v) = 0$. Otherwise, let $W(v) = W'(v)$.

Now we prove that the detector set D output by this algorithm is a weak (ϵ, k) detection set.

Let $\mathcal{C}$ be the set of connected components in $T \setminus D$. First, we note that every connected component in $\mathcal{C}$ has size at most α. If the weight were greater then α, look at the node in this component that is the ancestor of all other nodes in the tree-scan order. Now this node has weight greater than α, which would imply it would have been added to D. A contradiction!

We now show that D is a detection set. So, consider any subset $Z \subset V$ of size at most k. If $Z \cap D$ is non-empty, then we are done. So, assume that $Z \cap D = \emptyset$. Consider two disjoint sets C_1, C_2, each of size at least $\epsilon \cdot n$, in $G \setminus Z$. We will show that at least one of them contains a vertex of D, which is the witness to the failure.

If D intersects both C_1 and C_2, we are done. So, assume that C_1 has empty intersection with D. Because $|C_1| \geq \epsilon \cdot n$, there are at least $2k$ components $A_1, A_2, ..., A_{2k}$ of $\mathcal{C}$ that intersect C_1. In order to disconnect $(C_1 \cap A_i)$ from $G \setminus C_1$ without including any detectors, the adversary must delete at least one element (vertex or edge) from A_i. Hence, he must delete at least $2k > k$ vertices, which contradicts the definition of Z.