

ASSIGNMENT 3 - DISTRIBUTED ALGORITHMS

~~_____~~

- (1) (a) Prove that the analysis of the message complexity of *Dijkstra* algorithm that was studied in class is tight in the following sense: for arbitrary integers n and D , $1 \leq D \leq n$, there exists an n -vertex graph $G = (V, E)$ of diameter D , for which there is an execution of *Dijkstra* algorithm that requires $\Omega(n \cdot D + |E|)$ messages.
- (b) Repeat the previous section with the additional requirement that the graph G has $|E| = O(n)$ edges.

- (2) Give an example for an execution of the *Bellman - Ford* algorithm requiring $\Omega(n^3)$ messages.

(3)

Definition 0.1. Given a graph $G = (V, E)$ and a set of trees $\mathcal{T} = \{T_1, T_2, \dots, T_n\}$, where $\forall i \in \{1, 2, \dots, n\} : T_i = (V(T_i), E(T_i)), V(T_i) \subseteq V$ and $E(T_i) \subseteq E$, \mathcal{T} is called a *Cover* of G if the following conditions hold:

$$(a) \bigcup_{i=1}^n E(T_i) = E \text{ and } \bigcup_{i=1}^n V(T_i) = V.$$

(b) For each $e = (u, v) \in E$, there exists a tree $T_i \in \mathcal{T}$ so that $u, v \in V(T_i)$.

Definition 0.2. The *Overlap* of a *Cover* \mathcal{T} in a vertex $v \in V$ is the number of trees $T \in \mathcal{T}$ that contain v :

$$Overlap_v(\mathcal{T}) = |\{T \in \mathcal{T} \mid v \in V(T)\}|.$$

Definition 0.3. The *Overlap* of a *Cover* \mathcal{T} is the maximal *Overlap* of \mathcal{T} in some vertex $v \in V$:

$$Overlap(\mathcal{T}) = \max_{v \in V} \{Overlap_v(\mathcal{T})\}.$$

Definition 0.4. The *Diameter* of a *Cover* \mathcal{T} is the maximal *Diameter* of a tree within this *Coverage*:

$$Diam(\mathcal{T}) = \max_{T \in \mathcal{T}} \{Diam(T)\}.$$

Given a graph G and a *Cover* \mathcal{T} of G . Furthermore, each vertex $v \in V(G)$ knows to which trees $T \in \mathcal{T}$ it belongs, and the identity of his neighbours in each of these trees. Let $d = Diam(\mathcal{T})$ and $l = Overlap(\mathcal{T})$.

Describe a synchronizer that uses the *Cover* \mathcal{T} , and analyze its complexity. (Your analysis can rely on d and l). Of course, your synchronizer should be as effective as possible.

(4)

Definition 0.5. Given a graph $G = (V, E)$, a sub-graph $G' = (V, H)$, $H \subseteq E$, is considered a *k-spanning* graph of G if for each edge $e = (u, v) \in E$ there exist a path p in G' for which $|p| \leq k$. ($k \geq 1$)

Given a graph $G = (V, E)$ and a k -spanning sub-graph G' , where $k \geq 1$. Each vertex $v \in V$ knows the identity of its neighbours in G' , and it also knows that value of the constant k . Let $h = |E(G')|$.

Describe a synchronizer that uses G' and analyze its complexity. (Your analysis can rely on h). Of course, your synchronizer should be as effective as possible.

(5) Consider a 15-processor asynchronous network with the processors $\{P_0, P_1, \dots, P_{14}\}$. The processors constantly run a synchronizer. Let v and v' be two processors in the network, and suppose that at a certain moment, the pulse counter at v shows $p = 27$. What is the range of possible pulse numbers in at v' in each of the following cases:

- (a) The network is a ring (with the processors arranged according to their numbers), v is processor number 11, v' is processor number 2 and the synchronizer used is α .
- (b) The network is a full balanced binary tree (4 levels), v is the root, v' is one of the leaves and the synchronizer used is β .
- (c) The same as in (b), except both v and v' are leaves.