

Master2 Informatique UE: ALGORITHMIQUE DISTRIBUÉES – 4TIN913U Responsable : M. Gavoille Date : 4 janvier 2023 Durée : 1h30 Documents (course and personnal notes) allowed

Answers can be written in French or in English. In this examen, all graphs are supposed to be connected, even if not said explicitly.

## Bellman-Ford

In the course, we saw that the distributed Bellman-Ford algorithm computes, in the asynchronous case, a shortest-path spanning tree rooted at given vertex  $r_0$ . Remind that the principle of the algorithm is, for a vertex u, to send the length d of the current path to  $r_0$  to all its neighbors as soon as this distance decreases while updating its parent.

Recall that a graph is said to be *weighted* if each edge uv has a certain weight  $\omega(uv) \in \mathbb{R}^+$ . Obviously, a shortest-path spanning tree in a weighted graph must take into account the weight of its edges.

**Question 1.** Explain what is a "shortest-path spanning tree rooted at  $r_0$ " in a weighted graph G.

**Question 2.** Give a description of the distributed Bellman-Ford algorithm, i.e., the pseudo-code for any vertex u, when the graph is weighted. You will assume that the root  $r_0$  as well as  $\omega(uv)$  for each v-neighbor of u, are known from u.

We also saw that it was possible that, in some asynchronous scenario, Bellman-Ford produces  $2^{\Omega(n)}$  messages for a certain *n*-vertex weighted graph, i.e., at least  $2^{cn}$  messages when *n* is large enough and for a certain constant c > 0. The objective of the exercise is to show that in all cases, the algorithm cannot produce more than  $2^{O(n \log \Delta)}$  messages, where  $\Delta$  is the maximum degree of the graph.

Given a graph G, let p(G) be the maximum number of different paths of G that start from the same vertex. In this context, observe that: (1) a path cannot pass twice through the same vertex; (2) a path can consist of only one vertex and no edges; and (3) the number of paths may differ depending on the originating vertex. Note that for a cycle on three vertices u, v, w, the paths u - v - w and u - w - v are different.

**Question 3.** Compute  $p(G_0)$  in the case where  $G_0$  is the particular graph composed of two cycles on three vertices having one vertex in common. Specify the originating vertex realizing  $p(G_0)$ .

**Question 4.** Prove that, for any weighted graph G with n vertices, m edges and maximum degree  $\Delta$ , the distributed Bellman-Ford algorithm cannot produce more than  $\Delta + (2m - n) \cdot p(G)$  messages.

Let  $P(n, \Delta)$  be the maximum of p(G) for a graph G with n vertices and maximum degree  $\Delta$ .

**Question 5.** Prove that  $P(n, \Delta) \leq 1 + \Delta \sum_{i=0}^{n-2} (\Delta - 1)^i$ . [Hint: Consider the maximum number of paths, starting at the same vertex, and composed of exactly *i* edges.]

Let  $C_n$  denote the cycle graph on n vertices.

**Question 6.** Compute  $p(C_n)$  and show that the upper bound on P(n, 2) in Question 5 is optimal for  $\Delta = 2$ .

**Question 7.** Show that there exists a cubic graph G where  $p(G) < 1 + 3 \sum_{i=0}^{|V(G)|-2} 2^i$ . Recall that in a cubic graph, all the vertices have degree exactly three.

**Question 8.** From Question 4 and 5, show that for every weighted graph with n vertices and maximum degree  $\Delta$ , the distributed Bellman-Ford algorithm produces at most  $2^{O(n \log \Delta)}$  messages. [Hint:  $\Delta^n = 2^{n \log_2 \Delta}$ .]

**Question 9.** Why can we say that the counterexample of the course, showing a scenario and a weighted graph producing  $2^{\Omega(n)}$  messages is, in some way optimal? Discuss with pro and con arguments.

## Subgraph Detection

**Question 10.** Give an example of distributed task (i.e., a distributed problem) that is feasible in t rounds in the LOCAL model (for some  $t \in \mathbb{N}$ ), and not in the CONGEST model.

In the course, we studied distributed algorithms to detect cycles of length 3 or 4, namely  $C_3$  or  $C_4$ , in a graph G.

**Question 11.** More generally, for a fixed small graph H, explain what does mean "to detect a H in G" in the distributed setting. Specify the inputs and the outputs.

Let us define the *radius* of a graph G, denoted by  $\operatorname{rad}(G)$ , as the minimum depth of a spanning tree of G. In other words, it is the minimum eccentricity of G, that is  $\operatorname{rad}(G) = \min_{u \in V(G)} \{\operatorname{ecc}_G(u)\}$ . Here the edges have no weights, their cost is one.

**Question 12.** Give the radius of a  $3 \times 3$ -grid, the graph with 9 vertices obtained from a mesh of 3 rows and 3 columns. In addition to the value of the radius, draw the graph as well as a minimum depth spanning tree. What is the diameter of the graph? Same question for a  $4 \times 4$ -grid.

**Question 13.** Give a distributed algorithm, in the LOCAL model, to detect a  $C_5$  and specify its number of rounds (the smallest possible one).

**Question 14.** Give the principle of a distributed algorithm, in the LOCAL model, to detect a graph H that is supposed to be known from all the vertices. Specify the smallest possible number of rounds expressed as a fonction of rad(H).

We have seen in the course that every distributed algorithm in the *b*-CONGEST model that detects a  $C_4$  in a graph with *n* vertices requires  $\Omega(\sqrt{n}/b)$  rounds. Recall that it was partially based on the fact there are graphs without any  $C_4$  and with  $m \ge n^{1+1/2}$  edges. For the next question, whose goal is to generalize this result, we will assume that, for each integer  $k \ge 1$ , there is a graph  $G_k$  with *n* vertices and  $m \ge n^{1+1/k}$  edges without cycles of length 2k or less.

Question 15. Using an Alice-Bob simulation communication complexity argument based on the SET-DISJOINTNESS problem, explain why every distributed algorithm that detects a  $C_{2k}$  requires  $\Omega(n^{1/k}/b)$ rounds in the b-CONGEST model.