Theory of Distributed Systems

Winter Term 2021/22

Prof. Dr. Martin Hoefer
Marco Schmalhofer, Giovanna Varricchio

JOHANN WOLFGANG GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Institute of Computer Science
Algorithms und Complexity

Issued: 02.11.2021
Due: 09.11.2021, **8:15h**

Exercise 2

Please submit your solution in PDF format by sending an email to {schmalhofer,varricchio}@em.uni-frankfurt.de. Make sure that your solution reaches us before 8:15 am! Solutions are discussed on Nov 12th, 10:00h - 12:00h (Zoom Meeting-ID: 963 6309 6725, same password as lecture material).

Exercise 2.1. *Dijkstra* (6 = 2 + 2 + 2 Points)

Prove or disprove the following bounds:

For every n-vertex, D-diameter graph $G = (V, E)$, there exists an execution of Dijkstra's algorithm requiring

a) $\Omega(D^2)$ time,

b) $\Omega(|E|)$ messages,

c) $\Omega(nD)$ messages.

Exercise 2.2. *Bellman-Ford* (6 = 2 + 4 Points)

a) Give a tight upper bound to the message complexity of the Bellman-Ford algorithm if executed in the synchronous model.

b) Modify the Bellman-Ford algorithm (in the asynchronous model) so that it detects its termination. Make sure that the given bounds on time and message complexity still hold.

Exercise 2.3. *Synchronizers* (4 = 2 + 2 Points)

Given a network $G = (V, E)$, let us consider any two nodes u and v at pulses P_u and P_v, respectively. Show the following statements:

a) Running the synchronizer α, at any time it holds that $|P_v - P_u| \leq dist_G(u, v)$.

b) Running the synchronizer β, at any time it holds that $|P_v - P_u| \leq 1$.

Exercise 2.4. *Depth First Search (DFS)* (7 = 2 + 3 + 2 Points)

Consider a graph $G = (V, E)$ and a fixed node r_0. We have seen during lectures that in the synchronous CONGEST model the FLOOD algorithm builds a spanning tree of the network that, in particular, is a BFS-tree. Consider the following DFS algorithm in the synchronous CONGEST model:

Algorithm 1: DFS

```
T ← {r0}
each node marks all its neighbors as unexplored
r0 sends "join" to an unexplored neighbor u
repeat
    if node v gets "join" from u then
        if v ∈ T then
            v sends to u "ACK no parent"
            if u is unexplored for v then v marks u as explored
        else
            v becomes a leaf of T
            v sets Parent(v) ← u and marks u as explored
            if exists an unexplored neighbor w of v then v sends "join" to w else v sends "ACK
              parent" to Parent(v)
    if node v gets "ACK no parent" or "ACK parent" from u then
        v marks u as explored
        if exists an unexplored neighbor w of v then v sends "join" to w else v sends "ACK
          parent" to Parent(v)
until every node has marked all its neighbors as explored
```

a) Prove that both time and message complexity of the DFS algorithm are $\Theta(|E|)$.

b) Prove or disprove that the output of the algorithm is always a spanning tree of G rooted at r_0 with maximum depth among all the possible spanning trees of G rooted at r_0.

c) Describe an adjustment of the algorithm that decreases the time complexity to $\Theta(|V|)$.

The assignments and further information concerning the lecture can be found at
http://algo.cs.uni-frankfurt.de/lehre/tds/winter2122/tds2122.shtml

E-mail: schmalhofer@em.uni-frankfurt.de, varricchio@em.uni-frankfurt.de