School of Computer Science and Technology
University of Science and Technology of China

Exercise Sheet 6 for
Design and Analysis of Algorithms
Autumn 2022

Due 7 Jan 2023 at 16:59

SA22011035 Li Zhaoyi

Exercise 1 *40%*

Consider the synchronous network model for distributed algorithms. Let G be a graph with n nodes and m edges, and diameter D. Let s be a distinguished source (or root) node in G.

(a) Assume that the graph is unweighted, i.e., every edge has length 1. Give an algorithm such that after $O(D)$ rounds every node knows its distance to s.

(b) Assume that the graph is weighted with edge length in the range $1, 2, \cdots, k$. Give an algorithm that after $O(kD)$ rounds every node knows its distance to s. We do not assume that the edge lengths influence the time needed for sending messages. In each round, every node can send to each neighbor one message, regardless of the edge lengths.

You do not have to prove the correctness of your algorithms explicitly, but you should argue about their time complexity and message complexity.

Solution 1:

Algorithm of (a):

Algorithm 1: BFS-Shortest Path Algorithm

```
Input: Root UID i_0, Self UID i, Current Round r, Received Message message(d)
Output: Minimum Distance from i_0 to i
if r == 0 then
    marked(i) == False
    planned(i) == False
    if i == i_0 then
        dist = 0
    else
        dist = ∞
    end
if r == 1 then
    if i == i_0 then
        process i sends a search message search(dist) to its neighbors
    if message(d) is not None then
        marked(i) = True
        dist = 1, output dist
        planned(i) = True
else
    if planned(i) = True then
        process i sends a search message search(dist) to its neighbors
        planned(i) = False
    if marked(i) == False and message(d) is not None then
        marked(i) = True
        d ← decouple(message(d))
        dist = d + 1, output dist
        planned(i) = True
end
```

Time Complexity : Note that after the k_{th} ($k \geq 1$) round: if there exists a path of which the length $\leq k$, then we know that process i has already known its distance to process i_0 (source). So, given that the largest distance between any two nodes in this graph is D (for this graph is unweighted), we know that after at most $O(D)$ rounds, every node knows its distance to the source root s.

Message Complexity : Note that in our given algorithm, each node in our graph will emit and only emit message to all its neighbors once (for then it will be marked). So according to **Euler's Formula**:

$$\Sigma_{v \in V(G)} deg(v) = 2|E(G)|$$

the Message Complexity is $O(E)$.

Algorithm of (b):

Algorithm 2: Bellman-Ford Shortest Path Algorithm

Input: Root UID i_0, Self UID i, Current Round r, Received Message List *messages*
Output: Minimum Distance from i_0 to i

```
if r == 0 then
    if i == i0 then
        dist = 0
    else
        dist = ∞
    end
else
    process i0 sends a search message search(dist) to all its neighbors
    for j in range(len(messages)) do
        d_j ← decouple(messages[j])
        dist = min(dist, d_j + weight_ij)
    end
    if r == kD then
        output dist
end
```

(Note that in this exercise session, diameter refers to the number of edges rather than the real weighted distance.)

Time Complexity : Note that the following fact:
After the k_{th} ($k \geq 1$) round: If there exists a path of which the weighted length $\leq k$, then we induce that process i has already known its distance to process i_0 (source).
This is because the weight of any single edge is at least 1. So if the shortest path from s to process $i \leq k$, we can induce that there are at most $\frac{k}{1} = k$ edges in this shortest path and in this way we can claim that process i has already known its distance to s after the k_{th} round. So, given that there is at most D edges in the shortest path between any two nodes in this graph, we know that the weighted length of the shortest path is at most kD. So after at most $O(kD)$ rounds, every node knows its distance to the source root s.
Message Complexity : In each round, there are at most $O(E)$ messages being emited (Similarily, we leverage the **Euler Formula** and suppose that every node sends messages to all its neighbors). the Message Complexity is in total $O(E \cdot k \cdot D)$.

Exercise 2 *60%*
Read Chapter 1.1 and Chapter 2.1 in the book `https://people.csail.mit.edu/ghaffari/DA22/Notes/DGA.pdf` for the definitions of LOCAL model and CONGEST model.
Consider the synchronous distributed algorithms for Breadth-First Search Tree and the (Bellman-Ford) shortest paths from the class. Which algorithm(s) work in the LOCAL model and which work in the CONGEST model?

Solution 2:

My Conclusion:
Breadth-First Search Tree Algorithm can work in both of the **LOCAL** model and **CONGEST** model;
Bellman-Ford shortest path Algorithm can only work in the **LOCAL** model;

My Brief Justification for my conclusion:
First of all, according to the reading material, we can induce that the concept **CONGEST** model is constructed on the basis of the concept **LOCAL** model, for it requires additional constraint on the bandwith of the message.
For both of the BFS Algorithm and the Bellman-Ford Algorithm, according to the description of their algorithm and the definition of the **LOCAL** model, it is very straight that these two algorithms can work in the **LOCAL** model. Because each node in the given graph only need to know its unique identifier, n, source identifier (and might need to fetch the weight information). And Besides, each node sends messages to its neighbors, receives messages from its neighbors and eventually learns its own part of the output.
The BFS algorithm **can also** work in the **CONGEST** model. The Bellman-Ford algorithm **can not** work in the **CONGEST** model. Because as for the BFS algorithm, the **distance** information incorporated in the message can only discetely vary in the set $\{1, 2, ..., |V(G)| - 1, |V(G)|\}$, which means it has only $|V(G)|$ distinct options, and so its size can be bound in $O(\log |V(G)|)$. However, as for the Bellman-Ford algorithm, the

distance information incorporated in the message can nearly vary in the set $\{sum(E')|E' \subseteq E(G)\}$, where $sum(E')$ means the summation over the elements in E'. So apparently, its distance has $O(2^{V(G)})$ distinct options at most. To put it otherwise, the size of its message can only be bound in $O(\log 2^{|V(G)|}) = O(|V(G)|)$, which violate the definition of the **CONGEST** model.