PS Algorithmen für verteilte Systeme

https://avs.cs.sbg.ac.at/

Aufgabenblatt 2

Abgabe bis Donnerstag, 12.05.2022, 9:30 Uhr auf Blackboard.

Aufgabe 1 *Deutsch:* Zeigen Sie, dass im CONGEST Modell für ein beliebiges Netzwerk ein Leader in $O(D)$ Runden bestimmt werden kann, wobei D den Durchmesser des Netzwerks bezeichnet.

Anmerkung: Der Wert von D ist im CONGEST Modell kein initiales globales Wissen.

English: Show that, in the CONGEST model, we can appoint a leader in $O(D)$ rounds, where D is the diameter of the network.

Remark: The value of D is in the CONGEST model is initially not global knowledge.

Aufgabe 2

Deutsch: Gegeben sei folgender Algorithmus um einen Leader in einem synchronen, anonymen, non-uniformen Ring zu bestimmen:

1. Setze $L = V$
2. Knoten aus L erhalten ID 0 mit Wahrscheinlichkeit p und ansonsten ID 1
3. Knoten aus $V \setminus L$ erhalten ID 2
4. Führe mit diesen IDs Clockwise Algorithmus (mit Präferenz für kleinere IDs) aus
5. Setze L auf die Menge der vom Clockwise Algorithmus bestimmten Leader
6. Falls $|L| > 1$, wiederhole ab Schritt 2

Argumentieren Sie, dass – für eine geeignete Wahl einer Konstanten $p < 1$ (z.B. $p = \frac{1}{3}$) – dieser Algorithmus so implementiert werden kann, dass er mit hoher Wahrscheinlichkeit $O(n \log n)$ Runden benötigt und $O(n \log n)$ Nachrichten versendet. Sie dürfen den Algorithmus aus Aufgabe 3 von Aufgabenblatt 1 (herausfinden ob $|L| = 1$ für eine Menge Leaders L) als „Black Box“ verwenden.

Hinweis: Sie müssen insbesondere argumentieren, warum die allgemeine Schranke von $O(n^2)$ Nachrichten für den Clockwise Algorithmus aus der Vorlesung in dieser Anwendung zu pessimistisch ist.

Anmerkung: Der Algorithmus kann – mit ähnlichen Garantien – auch für asynchrone Ringe formuliert werden. In synchronen Ringen kann darüber hinaus argumentiert werden, dass jede gesendete Nachricht nur aus konstant vielen Bits besteht. Dadurch erhält man eine Schranke auf die *Gesamtgröße* aller gesendeten Nachrichten, die um einen Faktor von $\log n$ niedriger ist als die des Radius Growth Algorithmus, während die Rundenzahl um einen Faktor $\log n$ höher ist.

English Given is the following algorithm to elect a leader in a synchronous, anonymous, non-uniform ring:

1. Set $L = V$
2. Nodes from L receive ID 0 with probability p and otherwise ID 1
3. Nodes from $V \setminus L$ receive ID 2
4. Perform the clockwise algorithm with these IDs (with a preference for smaller IDs)
5. Set L as the set of leaders as outputted by the clockwise algorithm
6. If $|L| > 1$, repeat from step 2

Reason that – for a constant $p < 1$ chosen by you (z.B. $p = \frac{1}{3}$) – this algorithm can be implemented such that with high probability we need $O(n \log n)$ rounds and $O(n \log n)$ messages. You are allowed to use the algorithm from Exercise 3 of Aufgabenblatt 1 (determining whether $|L| = 1$ for a set L of leaders) as a „Black Box“.

Hint: In particular, you have to argue why the general bound of $O(n^2)$ messages is too pessimistic in this application.

Remark: The algorithm can also be formulated for asynnchronous rings – with similar guarantees. In synchronous rings, it can also be argued that every messages consists of at most a constant number of bits. Hence one can obtain a bound on the *total size* of all sent messages, which is a factor $\log n$ smaller than in the radius growth algorithm, while the number of rounds a factor of $\log n$ bigger is.

Aufgabe 3

Deutsch: Zeigen Sie, dass folgender Algorithmus einen $(2k-1)$-Spanner $H = (V, F)$ mit $O(n^{1+1/k})$ Kanten für einen Graph $G = (V, E)$ mit n Knoten berechnet:

1. $F = \emptyset$
2. Wähle beliebigen Knoten $s \in V$
3. Berechne Breitensuchbaum T von s in $G = (V, E)$ sowie $L_i(s) = \{v \in V \mid \mathrm{dist}_G(s, v) = i\}$ für alle $i \geq 0$

4. Sei $i(s)$ das kleinste i so dass $|L_i(s)| \leq |L_{i-1}(s)| \cdot n^{1/k}$
5. Füge für jeden Knoten in $L_0(s) \cup \cdots \cup L_{i(s)}(s)$ die Kante zum Elternknoten in T zu F hinzu
6. Entferne alle Knoten in $L_0(s) \cup \cdots \cup L_{i(s)-1}(s)$ aus V und entferne alle Kanten mit Endpunkten in $L_0(s) \cup \cdots \cup L_{i(s)-1}(s)$ aus E
7. Falls $V \neq \emptyset$, Wiederhole ab Schritt 2

Hinweis: Zeigen Sie zuerst, dass $i(s) \leq k$ für alle Startknoten s der Breitensuchbäume. Beachten Sie, dass für $L_i(s) = \emptyset$ die Ungleichung $|L_i(s)| \leq |L_{i-1}(s)| \cdot n^{1/k}$ trivialerweise gilt.

Anmerkung: Die Ungleichung $|L_i(s)| \leq |L_{i-1}(s)| \cdot n^{1/k}$ kann als Abbruchbedingung in der Durchführung der Breitensuche verwendet werden. Dadurch kann der Algorithmus im sequentiellen RAM Modell in linearer Zeit implementiert werden.

English: Show that, given a graph $G = (V, E)$ with n nodes, the following algorithm computes a $(2k-1)$-spanner $H = (V, F)$ with $O(n^{1+1/k})$ edges:

1. $F = \emptyset$
2. Chose any node $s \in V$
3. Compute a BFS tree T rooted at s in $G = (V, E)$ and $L_i(s) = \{v \in V \mid \mathrm{dist}_G(s, v) = i\}$ for every $i \geq 0$
4. Let $i(s)$ be the smallest i such that $|L_i(s)| \leq |L_{i-1}(s)| \cdot n^{1/k}$
5. For every node in $L_0(s) \cup \cdots \cup L_{i(s)}(s)$ add the edges to the parent node in T to F
6. Delete all nodes in $L_0(s) \cup \cdots \cup L_{i(s)-1}(s)$ from V and delete all edges with end points in $L_0(s) \cup \cdots \cup L_{i(s)-1}(s)$ from E
7. If $V \neq \emptyset$, repeat from step 2

Hint: As a first step, show that $i(s) \leq k$ for any root s of the BFS tree. Note that for $L_i(s) = \emptyset$ the inequality $|L_i(s)| \leq |L_{i-1}(s)| \cdot n^{1/k}$ holds trivially.

Remark: The inequality $|L_i(s)| \leq |L_{i-1}(s)| \cdot n^{1/k}$ can be used as a termination condition for the BFS algorithm. Hence the algorithm can be implemented in linear time in the sequential RAM model.

Bonusaufgabe 1

Deutsch: Zeigen Sie, dass in der asynchronen Variante des CONGEST Modells ein Breitensuchbaum in $O(D^2)$ Runden mit $O(m + nD)$ Nachrichten konstruiert werden kann, wobei n die Anzahl der Knoten, m die Anzahl der Kanten und den D den Durchmesser des Netzwerks bezeichnet.

English: Given is an asynchronous CONGEST network $G = (V, E)$ with n nodes, m edges, and diameter D. Show that we can construct a BFS tree within $O(D^2)$ rounds and $O(m + nD)$ messages.