Abteilung Algorithmik **Sommer ’12**
Institut für Betriebssysteme und Rechnerverbund
TU Braunschweig

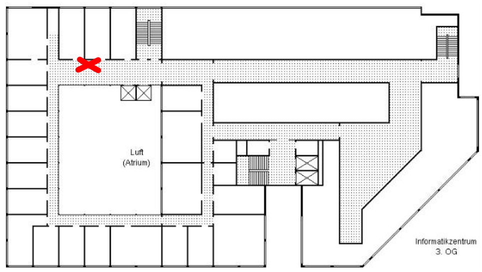
Alexander Kröller
Henning Hasemann
Stephan Friedrichs

Verteilte Algorithmen
Übung 3 vom 26. 5. 2012

Abgaben zu A und T am Dienstag, dem 12. 6. 2012, entweder

- vor der Vorlesung im IZ358, oder
- bis 9:40 im Hausaufgabenrückgabeschrank.

Bitte die Blätter vorne deutlich mit eigenem Namen versehen!
Abgaben zu P zum 12. 6. per Mail an `hasemann@ibr.cs.tu-bs.de`.



A — Allgemeiner Teil

Diese Aufgaben können von jedem bearbeitet werden, egal ob sich ansonsten für T oder P entschieden wird.

Aufgabe A1: Zeige, dass die Nachrichtenkomplexität des verteilten Dijkstra-Algorithmus (Alg. 2.10) nicht verbessert werden kann. Gib also für beliebiges n und $1 \leq D \leq n-1$ einen Graphen $G = (V, E)$ mit $|V| = n$ und Durchmesser D an, auf dem Algorithmus 2.10 $\Omega(nD + |E|)$ Nachrichten benötigt. **(3 P.)**

Aufgabe A2: Zeige: Im LOCAL-Modell mit IDs kann jedes Problem, das überhaupt verteilt gelöst werden kann, auch asynchron in Zeit $\mathcal{O}(n)$ mit $\mathcal{O}(n^2)$ Nachrichten gelöst werden. **(1 P.)**

T — Theoretischer Track

Diese beiden Aufgaben schliessen sich mit P aus:

Aufgabe T1: Gegeben sei ein Netzwerk, in dem jeder Knoten $v \in V$ einen Messwert $t_v \in \mathbb{R}$ (z.B. eine Temperatur) ermittelt hat. Ein Knoten $b \in V$ dient als Basisstation. Die Basisstation b möchte die k Knoten finden, deren Messungen vom allgemeinen Mittelwert am stärksten abweichen.
Wir gehen vom CONGEST-Modell aus, und nehmen an, dass ein Messwert in $\mathcal{O}(\log n)$ Bits codiert werden kann.

a) Gib einen Algorithmus an, mit dem b dieses Problem in Zeit $\mathcal{O}(\mathrm{diam}(G) + k)$ mit $\mathcal{O}(kn)$ Nachrichten löst. Beweise, dass Dein Algorithmus diese Komplexitäten erfüllt.

b) Beweise, dass $\Omega(\mathrm{diam}(G) + k)$ eine untere Schranke für die Laufzeit darstellt. (Hinweis: d.h., für jede Wahl von $\mathrm{diam}(G)$ und k gibt es Netzwerke, auf denen das Problem nicht schneller gelöst werden kann, sogar wenn mehr als $\mathcal{O}(kn)$ Nachrichten verschickt werden dürfen.)

c) Beweise, dass $\Omega(nk)$ eine untere Schranke für die Nachrichtenkomplexität darstellt. (Hinweis: d.h., für jedes n und $k < n$ gibt es Netzwerke, auf denen das Problem nicht mit weniger Nachrichten gelöst werden kann, sogar wenn die Laufzeit keine Rolle spielt.)

(2+2+2 P.)

P — Praktischer Track

Diese Aufgabe schliesst sich mit T aus. Sie läuft über **Blatt 2** und **Blatt 3** zusammen, ihr habt Zeit bis zum 12. 6.

Aufgabe P1: Entwirf und implementiere (in der Wiselib, klar) einen Algorithmus für Leader Election in allgemeinen Graphen, mit IDs.
Der Algorithmus soll mit dynamischen Netzen umgehen können, d.h.:

- Einige Knoten starten später („kommen dazu“),
- Knoten können sich beenden,
- Knoten könnnen sich bewegen und darüber ihre Nachbarschaft ändern,
- Durch die Bewegungen kann das Netzwerk geteilt bzw. wieder vereint werden.

Euer Algorithmus soll jeweils dafür sorgen, dass sich genau ein Knoten im Netz zum Leader erklärt. Ein geteiltes Netz mit mehreren Zusammenhangskomponenten ist dabei als mehrere Netze anzusehen. **(6+6 P.)**