PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 1

Abgabe bis Mittwoch, 18.03.2020, 11:00 Uhr auf https://abgaben.cosy.sbg.ac.at/

Aufgabe 1

Geben Sie eine Konfiguration der IDs in einem synchronen, nicht-anonymen, Ring mit n Knoten an, für die der Clockwise Algorithmus $\Theta(n)$ Nachrichten versendet und eine, für die der Clockwise Algorithmus $\Theta(n^2)$ Nachrichten versendet.

Aufgabe 2

Gegeben sei ein synchroner, anonymer, uniformer Ring mit n Knoten, in dem bereits ein Leader bestimmt wurde. Zeigen Sie, dass jedem Knoten in $O(n)$ Runden eine eindeutige ID zugeordnet werden kann.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 2

Abgabe bis Mittwoch, 25.03.2020, 11:00 Uhr auf `https://abgaben.cosy.sbg.ac.at/`

Aufgabe 3

Gegeben sei ein synchrones, anonymes, uniformes Netzwerk mit n Knoten, dessen Graph ein Baum (also zusammenhängend und kreisfrei) ist. Zeigen Sie, dass ein Leader durch einen randomisierten Algorithmus mit in Erwartung $O(n)$ Runden und $O(n)$ Nachrichten bestimmt werden kann.

Hinweis: Der Baum hat keine designierte Wurzel. Versuchen Sie das randomisierte Tie-Breaking möglichst einfach zu gestalten.

Aufgabe 4

Gegeben sei ein synchroner, anonymer, non-uniformer Ring mit n Knoten (in dem also n globales Wissen ist). Außerdem sei eine Teilmenge L der Knoten mit $|L| \geq 1$ gegeben, wobei in Bezug auf diese Menge jeder Knoten initial nur weiß, ob er selbst zu L gehört oder nicht. Zeigen Sie, dass in $O(n)$ Runden und mit $O(n)$ Nachrichten bestimmt werden kann, ob es sich um eine gültige Menge von Leadern mit $|L| = 1$ handelt. Das Ziel ist, dass am Ende alle Knoten wissen, ob das der Fall ist oder nicht.

Anmerkung: Der gesuchte Verifikationsalgorithmus kann – mit ähnlichen Garantien – auch für asynchrone Ringe formuliert werden. In synchronen Ringen kann zusätzlich noch der Fall $|L| = 0$ erkannt werden.

Bonusaufgabe 1

Zeigen Sie – mit möglichst elementaren Methoden – dass $(1 - \frac{1}{n})^{n-1} \geq \frac{1}{e}$ für $n \geq 2$ gilt (wobei e die Eulersche Zahl ist). Diese Schranke ist relevant, wenn im randomisierten Leader-Election-Algorithmus IDs aus dem Bereich von 0 bis $n - 1$ vergeben werden.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 3

Abgabe bis Mittwoch, 01.04.2020, 11:00 Uhr auf `https://abgaben.cosy.sbg.ac.at/`

Aufgabe 5

Zeigen Sie, dass im CONGEST Modell für ein beliebiges Netzwerk ein Leader in $O(D)$ Runden bestimmt werden kann, wobei D der Durchmesser des Netzwerks ist.

Aufgabe 6

Zeigen Sie, dass im CONGEST Modell der Durchmesser eines Netzwerks in $O(m)$ Runden berechnet werden kann, wobei m die Anzahl der Kanten des Netzwerks bezeichnet. Sie dürfen davon ausgehen, dass bereits ein Leader bestimmt wurde.

Hinweis: Gesucht ist eine generische Lösung, die für eine Vielzahl von Problemen funktioniert; die Berechnung des Durchmessers ist nur ein Beispiel von vielen. Im LOCAL Modell wären $O(D)$ Runden ausreichend.

Bonusaufgabe 2

In der Vorlesung haben wir den Queuing-Algorithmus für multiplen Upcast kennengelernt. Die Queue ist jedoch nicht unbedingt notwendig. Zeigen Sie, dass im CONGEST Modell für ein Netzwerk mit n Knoten und Durchmesser D ein multipler Upcast von k im Netzwerk verteilten Informationen mit jeweils Größe $O(\log n)$ an einen Wurzelknoten s über seinen gegebenen Breitensuchbaum in $O(k+D)$ Runden mit insgesamt $O(kD)$ Nachrichten durchgeführt werden kann, wenn jeder Knoten (zusätzlich zu den initial gespeicherten Informationen) nur Speicher proportional zur Anzahl seiner Nachbarn im Baum zur Verfügung hat.

Hinweis: Entwickeln Sie zunächst einen Algorithmus für den Spezialfall, dass sich alle Informationen an den Blättern des Breitensuchbaums befinden.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 4

Abgabe bis Mittwoch, 22.04.2020, 11:00 Uhr auf `https://abgaben.cosy.sbg.ac.at/`

Aufgabe 7

Zeigen Sie, dass im CONGEST Modell ein Maximal Independent Set in $O(D)$ Runden mit einem deterministischen Algorithmus bestimmt werden kann, wenn das Netzwerk ein Baum ist, wobei D den Durchmesser des Netzwerks bezeichnet.

Aufgabe 8

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 4 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.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 5

Abgabe bis Mittwoch, 29.04.2020, 11:00 Uhr auf https://abgaben.cosy.sbg.ac.at/

Aufgabe 9

In der Vorlesung haben wir nur Spanner-Konstruktionen für ungerichtete Graphen kennengelernt. Zeigen Sie, dass für *gerichtete* Graphen im Allgemeinen keine nicht-trivialen Spanner existieren, das heißt, dass es für jedes n einen gerichteten Graph mit n Knoten gibt, in dem jeder t-Spanner für $t < n$ mindestens $\Omega(n^2)$ Kanten hat.

Hinweis: Sie müssen nicht davon ausgehen, dass der Ausgangsgraph stark zusammenhängend ist, d. h., es darf Knoten u und v geben, für die es keinen Pfad von u nach v gibt (also $\mathrm{dist}(u, v) = \infty$*).*

Aufgabe 10

In der Vorlesung haben wir einen Greedy-Algorithmus kennengelernt, der für eine beliebig gewählte Ganzzahl $k \geq 2$ einen $(2k - 1)$-Spanner eines gegebenen ungerichteten, ungewichteten Graphen berechnet. Zeigen Sie, dass mit einer Modifikation des Greedy-Algorithmus für jeden gegebenen ungerichteten, *gewichteten* Graph ein $(2k - 1)$-Spanner mit $O(n^{1+1/k})$ Kanten konstruiert werden kann.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 6

Abgabe bis Mittwoch, 06.05.2020, 11:00 Uhr auf https://abgaben.cosy.sbg.ac.at/

Aufgabe 11

Zeigen Sie, dass im CONGEST Modell für jedes Netzwerk $G = (V, E)$ mit n Knoten aus einem gegebenen (ρ, μ, ℓ)-Cover in konstant vielen Runden ein $(2\rho + 1)$-Spanner von G mit $O(\mu + \ell n)$ Kanten konstruiert werden kann, wenn jeder Knoten für jedes Cluster, in dem er enthalten ist, seinen Parent und seine Kinder im Baum des Clusters sowie die IDs des Clusterzentrums kennt.

Aufgabe 12

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

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.

Bonusaufgabe 3

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.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 8

Abgabe bis Mittwoch, 13.05.2020, 11:00 Uhr auf https://abgaben.cosy.sbg.ac.at/

Aufgabe 13

Zeigen Sie, dass im CONGEST Modell eine 2-Approximation $\hat{D}$ des Durchmessers D des Netzwerks in $O(D)$ Runden bestimmt werden kann. Gesucht ist also eine Zahl $\hat{D}$, so dass $\frac{1}{2}D \leq \hat{D} \leq D$. Sie dürfen davon ausgehen, dass bereits ein Leader bestimmt wurde.

Hinweis: Die Dreiecksungleichung besagt, dass $\operatorname{dist}(u, v) \leq \operatorname{dist}(u, w) + \operatorname{dist}(w, v)$ für alle Knoten u, v und w.

Aufgabe 14

Zeigen Sie, dass im CONGEST Modell eine $O(\log n)$-Approximation des APSP Problems für ungewichtete Graphen mit n Knoten durch einen Las-Vegas Algorithmus mit erwarteter Rundenzahl $O(n \log n)$ berechnet werden kann. Jeder Knoten u soll also am Ende für jeden anderen Knoten v eine Zahl $\delta(u, v)$ kennen, so dass $\operatorname{dist}(u, v) \leq \delta(u, v) \leq O(\log n) \cdot \operatorname{dist}(u, v)$. (Sie können davon ausgehen, dass bereits ein Leader bestimmt wurde.)

Hinweis: In der Vorlesung wurde nur ein Monte Carlo-*Algorithmus für die* exakte *APSP-Berechnung vorgestellt. Es gibt mehrere Lösungsansätze für diese Aufgabe, insbesondere wäre es möglich den Durchmesser in der gefordeten erwarteten Laufzeit auch exakt zu berechnen.*

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 8

Abgabe bis Mittwoch, 20.05.2020, 11:00 Uhr auf https://abgaben.cosy.sbg.ac.at/

Aufgabe 15

Gegeben sei ein (Monte-Carlo) Algorithmus für das (exakte) SSSP Problem für gewichtete Graphen, der im CONGEST Modell mit hoher Wahrscheinlichkeit eine korrekte Lösung berechnet (und mit geringer Wahrscheinlichkeit falsche Distanzwerte berechnet). Zeigen Sie, dass die Korrektheit einer Ausgabe dieses Algorithmus (bestehend aus einer Distanz für jeden Knoten) in $O(D)$ Runden verifiziert werden kann.

Hinweis: Man kann eine lokale „Optimalitätsbedingung“ angeben.

Aufgabe 16

Zeigen Sie, dass im CONGEST Modell eine $(1 + \epsilon)$-Approximation für das SSSP Problem für gewichtete Graphen mit n Knoten in $O(\sqrt{nD}\log^2(n)/\epsilon)$ Runden berechnet werden kann (wenn das höchste Kantengewicht W polynomiell in n ist, also $W = n^{o(1)}$.

Hinweis: „Simulieren“ Sie eine geeignete Variante von Dijkstras Algorithmus auf dem Overlay Netzwerk $H = (Z, Z \times Z)$, dessen Knoten die Zentren sind und dessen Kantengewichte den approximativen h-Distanzen zwischen den Zentren entsprechen.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 9

Abgabe bis Mittwoch, 27.05.2020, 11:00 Uhr auf `https://abgaben.cosy.sbg.ac.at/`

Aufgabe 17

Gegeben sei ein Las-Vegas-Algorithmus für ein Problem P im CONGEST Modell mit einer (allen Knoten explizit bekannten) erwarteten Laufzeit von $R(n)$ Runden für ein Netzwerk mit n Knoten. Zeigen Sie, dass es für ein Netzwerk mit n Knoten und Durchmesser D einen Monte-Carlo-Algorithmus für P im CONGEST Modell gibt, der, für jedes gegebene $c \geq 1$, immer Laufzeit $O((R(n) + D) \cdot c \log n)$ hat und mit Wahrscheinlichkeit mindestens $1 - \frac{1}{n^c}$ korrekt ist.

Hinweis: Markov Bound

Aufgabe 18

Gegeben sei ein beliebiges Entscheidungsproblem P (d.h. es gibt nur Ausgaben der Form YES oder NO). Angenommen, wir haben einen randomisierten Algorithmus $\mathscr{A}$ für P mit den folgenden Eigenschaften:

- Für alle Eingaben $x \in P$ gilt $\Pr[\mathscr{A}(x) = \mathsf{NO}] \leq 1/3$ und
- für alle Eingaben $x \notin P$ gilt $\Pr[\mathscr{A}(x) = \mathsf{YES}] \leq 1/3$.

Zeigen Sie, dass man durch logarithmisch viele Wiederholungen von $\mathscr{A}$ einen Algorithmus für das Problem P mit Fehlerwahrscheinlichkeit $1/n^c$ (für eine beliebige vorgegebene Konstante c) erhalten kann.

Hinweis: Chernoff Bound

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 10

Abgabe bis Mittwoch, 03.06.2020, 11:00 Uhr auf `https://abgaben.cosy.sbg.ac.at/`

Aufgabe 19

In der Vorlesung analysieren wir den Prozess der epidemischen Informationsausbreitung nur für vollständige Graphen. Dabei werden sowohl im Push- als auch im Pull-Modell mit hoher Wahrscheinlichkeit $\Theta(\log n)$ viele Runden benötigt bis alle n Knoten des Netzwerks infiziert sind. Zeigen Sie, anhand einer geeigneten Klasse von Beispielgraphen, dass sich für allgemeine Graphen mit n Knoten bei ungünstiger Startkonfiguration die Anzahl der benötigten Runden im Push- und Pull-Modell um *mehr* als einen konstanten Faktor unterscheiden kann.

Aufgabe 20

In der Vorlesung wurde bewiesen, dass während der Wachstumsphase im Push-Modell der Wachstumsfaktor für die Anzahl der infizierten Knoten mit Wahrscheinlichkeit höchstens $\frac{1}{e^{1/24}}$ höchstens $\frac{7}{6}$ beträgt, also dass $\Pr[I(t+1) \leq \frac{7}{6} I(t)] \leq \frac{1}{e^{1/24}}$ unter der Voraussetzung $I(t) \leq \frac{n}{3}$ gilt. Zeigen Sie, ausgehend von dieser Ungleichung, mittels Anwendung der Chernoff-Bound, dass die Wachstumsphase aus $O(\log n)$ Runden besteht.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 11

Abgabe bis Mittwoch, 10.06.2020, 11:00 Uhr auf `https://abgaben.cosy.sbg.ac.at/`

Aufgabe 21

Zeigen Sie, dass im Push-Modell für n Knoten folgendes gilt: Wenn $c' \ln n \leq G(t) \leq \frac{2}{3}n$ für eine passende Konstante c' gilt, dann ist $G(t+1) \leq 0.9 \cdot G(t)$ mit hoher Wahrscheinlichkeit (also mit Wahrscheinlichkeit mindestens $1 - \frac{1}{n^c}$ für eine vorgegebene Konstante c).

Hinweis: Die Aussage gilt jedenfalls für $c' = 288c$.

Aufgabe 22

Zeigen Sie, dass im Pull-Modell für n Knoten, ausgehend von einem infizierten Knoten, nach $O((\log n)^2)$ Runden mit hoher Wahrscheinlichkeit mindestens $\frac{n}{\ln n}$ Knoten infiziert sind.

Anmerkung: Diese Aufgabe vervollständigt die Analyse des Pull-Modells, die wir in der Vorlesung nur für den Fall $I(t) > \frac{n}{\ln n}$ durchgeführt haben. Es gibt mehrere Wege, diese Aufgabe zu lösen, und insbesondere ist es mit einfachen Mitteln möglich, eine bessere Schranke von $O(\log n)$ zu zeigen. Der Faktor $\frac{1}{\ln n}$ ist für die Analyse nicht zentral; um die Notation möglichst einfach zu halten, kann es hilfreich sein, auf die etwas stärkere Garantie abzuzielen, mindestens $\frac{n}{3}$ Knoten zu infizieren.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt 12

Abgabe bis Mittwoch, 17.06.2020, 11:00 Uhr auf `https://abgaben.cosy.sbg.ac.at/`

Aufgabe 23

Schreiben Sie ein Programm, das den Push-Algorithmus auf einem vollständigen Graph mit n Knoten simuliert. Ihr Programm muss nicht verteilt oder parallel laufen, es soll den Algorithmus lediglich simulieren. Geben Sie für jedes $n \in \{10, 100, 1000, 10000\}$ den Mittelwert der Laufzeiten von 50 unabhängigen Simulationsdurchläufen aus, das heißt die durchschnittliche Anzahl der Runden, die benötigt wird, um alle Knoten zu infizieren (ausgehend von einem infizierten Knoten). Ermitteln Sie anschließend diejenige Konstante c, für die der empirisch gemessene Mittelwert der Laufzeiten (näherungsweise) als $c \ln n$ ausgedrückt werden kann. Bitte gestalten Sie Ihre Abgabe so, dass sie den Quelltext, die Simulationsergebnisse und Ihre Berechnungen enthält.

Hinweis: Achten Sie bei sequentieller Ausführung darauf, dass in jeder Runde jeder gesunde Knoten nur von denjenigen Knoten infiziert werden kann, die bereits am Anfang der Runde infiziert waren.

Aufgabe 24

Angenommen, eine Münze wird n Mal hintereinander geworfen. Sei, für jedes $1 \leq i \leq n$, X_i die binäre Zufallsvariable, die 1 ist, wenn der i-te Münzwurf Kopf zeigt, und ansonsten 0. Weiters sei $X := \sum_{i=1}^{n} X_i$ die Zufallsvariable, die angibt, wie oft die Münze bei n Würfen Kopf zeigt, und sei A das Ereignis, dass X seinen Erwartungswert um mindestens 5 Prozent übersteigt.

Schreiben Sie ein Computerprogramm in einer Programmiersprache Ihrer Wahl, das n Münzwürfe simuliert. Wiederholen Sie diese Simulation für jedes $n \in \{10, 100, 1000, 10000\}$ jeweils 50 Mal und bestimmen Sie für jedes n die relative Häufigkeit des Ereignisses A, also den relativen Anteil an Simulationswiederholungen, in denen X stark von seinem Erwartungswert abweicht. Vergleichen Sie anschließend für jedes $n \in \{10, 100, 1000, 10000\}$ (a) die Chernoff-Schranke für die Wahrscheinlichkeit des Ereignisses A mit (b) der empirisch ermittelten relativen Häufigkeit von A. Führen Sie Ihre Simulation mit zwei verschiedenen Zufallsgeneratoren durch, etwa `rnd` und `random.SystemRandom()` in Python oder `java.util.Random` und `java.security.SecureRandom` in Java, und erläutern Sie mit welchem Zufallsgenerator das Ereignis A häufiger auftritt. Bitte gestalten Sie Ihre Abgabe so, dass sie den Quelltext, die Simulationsergebnisse und Ihre Berechnungen enthält.

Bonusaufgabe 4

Zeigen Sie, dass es in einem *anyonymen* Ring unmöglich ist, ein Maximal Independent Set zu berechnen (auch nicht in einem synchronen, non-uniformen Ring).

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt A

Abgabe bis Mittwoch, 29.04.2020, 11:00 Uhr per E-Mail an forster@cs.sbg.ac.at

Aufgabe 1

Gegeben sei ein synchroner, anonymer, non-uniformer Ring mit n Knoten (in dem also n globales Wissen ist). Außerdem sei eine Teilmenge L der Knoten mit $|L| \geq 1$ gegeben, wobei in Bezug auf diese Menge jeder Knoten initial nur weiß, ob er selbst zu L gehört oder nicht. Zeigen Sie, dass in $O(n)$ Runden und mit $O(n)$ Nachrichten bestimmt werden kann, ob es sich um eine gültige Menge von Leadern mit $|L| = 1$ handelt. Das Ziel ist, dass am Ende alle Knoten wissen, ob das der Fall ist oder nicht.

Anmerkung: Der gesuchte Verifikationsalgorithmus kann – mit ähnlichen Garantien – auch für asynchrone Ringe formuliert werden. In synchronen Ringen kann zusätzlich noch der Fall $|L| = 0$ erkannt werden.

Aufgabe 2

Zeigen Sie, dass im CONGEST Modell für ein beliebiges Netzwerk mit n Knoten ein Leader in $O(n)$ Runden bestimmt werden kann.

Aufgabe 3

Gegeben sei ein Las-Vegas-Algorithmus für ein Problem P mit einer (allen Knoten explizit bekannten) erwarteten Laufzeit von $R(n)$ Runden für ein Netzwerk mit n Knoten. Zeigen Sie, dass es für ein Netzwerk mit n Knoten und Durchmesser D einen Monte-Carlo-Algorithmus für P gibt, der, für jedes $c \geq 1$ immer Laufzeit $O((R(n) + D) \cdot c \log n)$ hat und mit Wahrscheinlichkeit mindestens $1 - \frac{1}{n^c}$ korrekt ist.

PS Algorithmen für verteilte Systeme

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

Aufgabenblatt B

Abgabe bis Mittwoch, 10.06.2020, 11:00 Uhr per E-Mail an forster@cs.sbg.ac.at

Aufgabe 4

In der Vorlesung haben wir nur Spanner-Konstruktionen für ungerichtete Graphen kennengelernt. Zeigen Sie, dass für *gerichtete* Graphen im Allgemeinen keine nicht-trivialen Spanner existieren, das heißt, dass es für jedes n einen gerichteten Graph mit n Knoten gibt, in dem jeder t-Spanner für $t < n$ mindestens $\Omega(n^2)$ Kanten hat.

Hinweis: Sie müssen nicht davon ausgehen, dass der Ausgangsgraph stark zusammenhängend ist, d. h., es darf Knoten u und v geben, für die es keinen Pfad von u nach v gibt (also $\mathrm{dist}(u, v) = \infty$).

Aufgabe 5

Zeigen Sie, dass im CONGEST Modell eine 2-Approximation $\hat{D}$ des (ungewichteten) Durchmessers D des Netzwerks in $O(D)$ Runden bestimmt werden kann. Gesucht ist also eine Zahl $\hat{D}$, so dass $\frac{1}{2}D \leq \hat{D} \leq D$. Sie dürfen davon ausgehen, dass bereits ein Leader bestimmt wurde.

Hinweis: Die Dreiecksungleichung besagt, dass $\mathrm{dist}(u, v) \leq \mathrm{dist}(u, w) + \mathrm{dist}(w, v)$ für alle Knoten u, v und w.

Aufgabe 6

Zeigen Sie, dass in Phase 3 des Push-Modells folgendes gilt: Wenn $G(t) \geq 288c \ln n$, dann ist $G(t + 1) \leq 0.9 \cdot G(t)$ mit hoher Wahrscheinlichkeit (also mit Wahrscheinlichkeit mindestens $1 - \frac{1}{n^c}$).