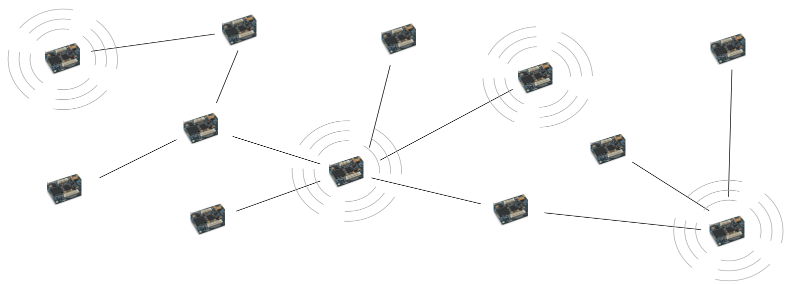
Algorithmen für Ad-hoc- und Sensornetze

VL 10 – Eine kurze Geschichte vom Färben

Markus Völker | 27. Juni 2012 (Version 1)

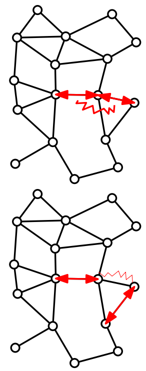
INSTITUT FÜR THEORETISCHE INFORMATIK - LEHRSTUHL FÜR ALGORITHMIK (PROF. WAGNER)



Motivation

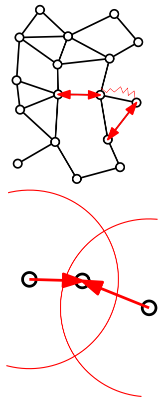
- Kommunikation im drahtlosen Kanal ist nicht beliebig gleichzeitig möglich
  - kein Knoten kann gleichzeitig an zwei Übertragungen teilnehmen
  - benachbarte Knoten können häufig nicht gleichzeitig übertragen/empfangen

⇒ Einfach Drauflossenden ist keine Lösung!



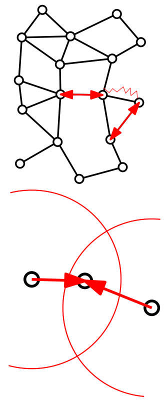
Ad-hoc-Lösung: CSMA/CD

- Ad-hoc-Lösung: CSMA/CD (wie Ethernet)
  - *Carrier Sense Multiple Access/Collision Detection*
  - Carrier Sense: Mithören, ob der Kanal frei ist
  - Collision Detection: Bei Kollision später wiederholen
  - Achtung: Anders als im Ethernet kann der Sender Kollisionen nicht immer erkennen!
  - Was, wenn der Empfänger von jemandem gestört wird, den der Sender nicht sieht? (Hidden Terminal Problem)



Ad-hoc-Lösung: CSMA/CA

- Ad-hoc-Lösung: CSMA/CA (ähnlich Ethernet)
  - *Carrier Sense Multiple Access/Collision Avoidance*
  - Carrier Sense: Mithören, ob der Kanal frei ist
  - Collision Avoidance: Kollisionen verhindern
  - Achtung: Anders als im Ethernet kann der Sender Kollisionen nicht immer erkennen!
  - Was, wenn der Empfänger von jemandem gestört wird, den der Sender nicht sieht? (Hidden Terminal Problem)
- CSMA/CA-Protokolle (gaaaaanz grob)
  - Sender wartet, bis er freien Kanal sieht
  - Sender schickt „Request To Send"
  - Empfänger bestätigt „Clear To Send"
    - (das hören hoffentlich auch alle Betroffenen!)
  - Sender schickt Daten
  - Empfänger bestätigt



Die koordinierte Lösung: TDMA

- Übertragungen bekommen Slots in Zeitraster zugewiesen
  - *Time Division Multiple Access*
  - entweder kurzfristig bei Bedarf oder sogar langfristig/periodisch
- Übertragungen sind auch *gleichzeitig* möglich, wenn mehrere Frequenzen/orthogonale Kodierungen zur Verfügung stehen
  - CDMA: Code Division ..
  - FDMA: Frequency Division ..

Übertragungen können sogar gleichzeitig dieselbe Ressource belegen, wenn sie weit genug auseinander liegen!

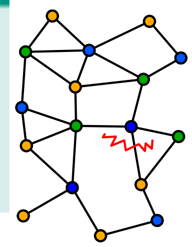
Knotenfärbung

Definition

Eine Knotenfärbung eines Graphen $G = (V, E)$ ist eine Abbildung $c : V \to \mathbb{N}$ so, dass für jede Kante die beiden Endpunkte unterschiedliche Farben haben, also

$$\{u, v\} \in E \Rightarrow c(u) \neq c(v)$$

- Eine Färbung c hat die Größe $|c| = \max_{v \in V} c(v)$



- Jeder darf nur senden, wenn „seine Farbe" dran ist
- Eigentlich bringt das gar nichts

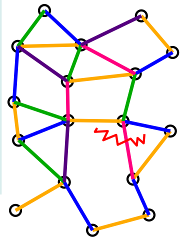
Kantenfärbung

Definition

Eine Kantenfärbung eines Graphen $G = (V, E)$ ist eine Abbildung $c : E \to \mathbb{N}$ so, dass für jeden Knoten alle inzidenten Kanten unterschiedliche Farben haben, also

$$\{u, v\}, \{u, w\} \in E \Rightarrow c(\{u, v\}) \neq c(\{u, w\})$$

- Eine Färbung c hat die Größe $|c| = \max_{e \in E} c(e)$



- Jeder darf nur über Kante kommunizieren, „deren Farbe" dran ist
- schließt zumindest direkte Kollisionen aus

Knoten-, Kanten-, Abstand-d-Färbungen

Beobachtung

Eine Kantenfärbung ist eine Knotenfärbung im Graphen $G' = (E, E')$ mit $E' := \{\{e_1, e_2\} : e_1 \cap e_2 \neq \emptyset\}$

- Wir können Kantenfärbungen auf Knotenfärbungen zurückführen!
- das lässt sich erweitern auf andere Färbungen
  - Abstand-d-Färbungen: Knoten mit Abstand $\leq d$ müssen unterschiedlich gefärbt werden
  - jede Bedingung, in der Ausschlüsse innerhalb von konstanter Entfernung liegen!

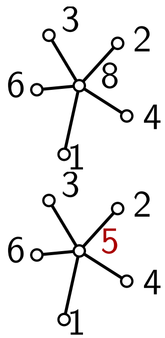
Deshalb kümmern wir uns nur um Knotenfärbungen!

Warm-up: Zentrale $\Delta + 1$-Färbung

Satz

Jeder Graph lässt sich mit $\Delta + 1$ Farben färben.

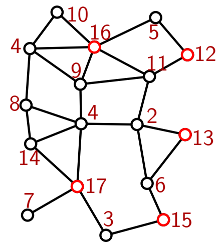
- Beweis: wenn es einen Knoten u gibt, der eine höhere Farbe hat als $\deg(u) + 1$, dann ist eine der „Farben“ $1, \ldots, \deg(u) + 1$ in der Nachbarschaft nicht verwendet worden
  - $\Rightarrow$ Diese Farbe kann der Knoten verwenden
- besser wollen wir gar nicht sein, aber wie bekommen wir das *verteilt* hin?



Warm-up: Verteilte $\Delta + 1$-Färbung

Einfache Lösung

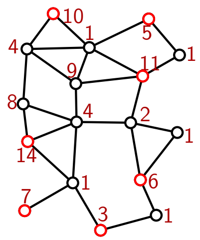
1. jeder färbt sich mit seiner ID
2. jeder Knoten u mit Farbe $c(u) > \deg(u) + 1$, der die höchste Farbe in seiner Nachbarschaft hat, wählt die kleinste Farbe aus $1, \ldots, \deg(u) + 1$ aus, die keiner der Nachbarn hat



Warm-up: Verteilte $\Delta + 1$-Färbung

Einfache Lösung

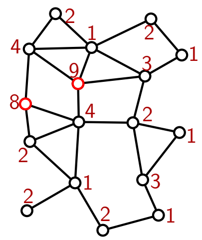
1. jeder färbt sich mit seiner ID
2. jeder Knoten u mit Farbe $c(u) > \deg(u) + 1$, der die höchste Farbe in seiner Nachbarschaft hat, wählt die kleinste Farbe aus $1, \ldots, \deg(u) + 1$ aus, die keiner der Nachbarn hat



Warm-up: Verteilte $\Delta + 1$-Färbung

Einfache Lösung

1. jeder färbt sich mit seiner ID
2. jeder Knoten u mit Farbe $c(u) > \deg(u) + 1$, der die höchste Farbe in seiner Nachbarschaft hat, wählt die kleinste Farbe aus $1, \ldots, \deg(u) + 1$ aus, die keiner der Nachbarn hat



Ausblick nächste Vorlesung

Satz

Es gibt einen verteilten Algorithmus, der eine $\Delta + 1$-Färbung in $O(\Delta^2 + \log_2^*(n))$ Schritten berechnet.

- $\log^*(n)$: Anzahl der Anwendungen von log, bevor das Argument unter 1 fällt. (Beispiel: $\log_2^*(10^{50}) = 5$)
- Andere Möglichkeit sich das zu merken:

$$\log_2^*(x) = 5 \Leftrightarrow 2^{2^{2^{2}}} < x \leq 2^{2^{2^{2^{2}}}}$$

Literatur

1. A. Goldberg, S. Plotkin, G. Shannon: *Parallel symmetry-breaking in sparse graphs*. In: Proceedings of the nineteenth annual ACM symposium on Theory of computing, 1987.