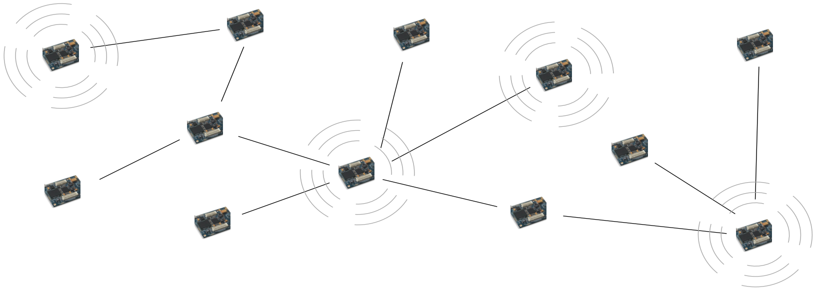
Algorithmen für Ad-hoc- und Sensornetze

Übung 6 – Kommunikation und Färbungen im SINR Modell (basierend auf VL11)

Fabian Fuchs | 17. Jan. 2015 (Version 1)

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



Motivation

Interferenzmodell bisher:

- kein Knoten kann gleichzeitig an zwei Übertragungen teilnehmen
- benachbarte Knoten können nicht gleichzeitig übertragen/empfangen

Realität für Drahtloskommunikation ist komplexer!

- Globale Interferenz kann Übertragungen verhindern

Wiederholung: SINR

SINR-Modell, allgemein

Im *Signal-to-Interference-plus-Noise-Ratio Modell* kann ein Empfänger r eine Nachricht eines Senders s dekodieren, wenn

$$\frac{P/\operatorname{dist}(s,r)^\alpha}{\sum_{s' \neq s} P/\operatorname{dist}(s',r)^\alpha + N} \geq \beta$$

- P: Signalstärke, mit der s sendet
- $\operatorname{dist}(s,r)^\alpha$: *geometrischer* Leistungsabfall zwischen s und r

SINR-Modell ist sehr viel realistischer, aber gleichzeitig viel komplexer zu analysieren.

Wiederholung: SINR

SINR-Modell, allgemein

Im *Signal-to-Interference-plus-Noise-Ratio Modell* kann ein Empfänger r eine Nachricht eines Senders s dekodieren, wenn

$$\frac{P/\operatorname{dist}(s,r)^\alpha}{\sum_{s' \neq s} P/\operatorname{dist}(s',r)^\alpha + N} \geq \beta$$

- P: Signalstärke, mit der s sendet
- $\operatorname{dist}(s,r)^\alpha$: *geometrischer* Leistungsabfall zwischen s und r

SINR-Modell ist sehr viel realistischer, aber gleichzeitig viel komplexer zu analysieren.

Wahrscheinlichkeitstheorie

Sei n die Anzahl an Knoten im Netzwerk.

Definition: mit hoher Wahrscheinlichkeit

Ein Ereignis geschieht mit hoher Wahrscheinlichkeit, wenn die Wahrscheinlichkeit dafür mindestens $(1 - \frac{1}{n})$ ist.

- Damit ist die Fehlerwahrscheinlichkeit $< \frac{1}{n}$

Wahrscheinlichkeitstheorie

Sei n die Anzahl an Knoten im Netzwerk.

Definition: mit hoher Wahrscheinlichkeit

Ein Ereignis geschieht mit hoher Wahrscheinlichkeit, wenn die Wahrscheinlichkeit dafür mindestens $(1 - \frac{1}{n})$ ist.

- Damit ist die Fehlerwahrscheinlichkeit $< \frac{1}{n}$

Satz (o.B.)

Tritt ein Ereignis in jeder Runde mit konstanter Wahrscheinlichkeit $\frac{1}{c}$ ein, ist die Wahrscheinlichkeit das es nach $c \cdot \log_e n$ Runden einmal eingetreten ist mindestens $1 - \frac{1}{n}$.

Wahrscheinlichkeitstheorie

Sei n die Anzahl an Knoten im Netzwerk.

Definition: mit hoher Wahrscheinlichkeit

Ein Ereignis geschieht mit hoher Wahrscheinlichkeit, wenn die Wahrscheinlichkeit dafür mindestens $(1 - \frac{1}{n})$ ist.

- Damit ist die Fehlerwahrscheinlichkeit $< \frac{1}{n}$

Satz (o.B.)

Tritt ein Ereignis in jeder Runde mit konstanter Wahrscheinlichkeit $\frac{1}{c}$ ein, ist die Wahrscheinlichkeit das es nach $c \cdot \log_e n$ Runden einmal eingetreten ist mindestens $1 - \frac{1}{n}$.

Idee: Eintrittswahrscheinlichkeit ist mindestens

$$1 - \underbrace{\left(1 - \frac{1}{c}\right)^{c \cdot \log_e n}}_{FehlerWS'} \geq 1 - \left(1 - \frac{\log_e n}{c \log_e n}\right)^{c \cdot \log_e n} \overset{*(1)}{\geq} 1 - e^{-\log_e n} = 1 - 1/n$$

(1) gilt da $(1 + \frac{t}{x})^x \leq e^t$)

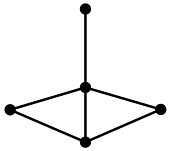
Local Broadcasting

Grundlegende Fragestellung: Wann soll welcher Knoten senden?

Local Broadcasting Problem

Jeder Knoten im Netzwerk sendet eine Nachricht an seine Nachbarn

- Knoten innerhalb der Sendereichweite müssen erreicht werden



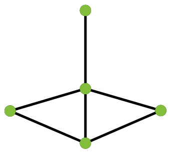
Local Broadcasting

Grundlegende Fragestellung: Wann soll welcher Knoten senden?

Local Broadcasting Problem

Jeder Knoten im Netzwerk sendet eine Nachricht an seine Nachbarn

- Knoten innerhalb der Sendereichweite müssen erreicht werden
- Im Beispiel: Nacheinander ($\Rightarrow \Omega(n)$ Zeitslots)
- In Praxis: Möglichst viele Knoten gleichzeitig
- Ziel: Minimiere Laufzeit



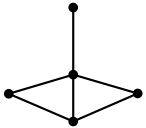
Local Broadcasting kann in $O(\Delta \log n)$ Zeit erreicht werden!

Local Broadcasting Algorithmus

Local Broadcasting

Jeder Knoten kennt n und Maximalgrad Δ

1. Sende in jedem Zeitslot mit Wahrscheinlichkeit $1/O(\Delta)$

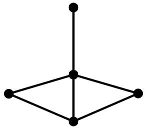


Local Broadcasting Algorithmus

Local Broadcasting

Jeder Knoten kennt n und Maximalgrad Δ

1. Sende in jedem Zeitslot mit Wahrscheinlichkeit $1/O(\Delta)$



Satz

Nach $O(\Delta \log n)$ Zeitslots hatte jeder Knoten mit *hoher* Wahrscheinlichkeit ($> 1 - 1/n$) Erfolg.

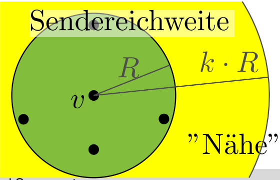
Local Broadcasting Algorithmus

Satz

Nach $O(\Delta \log n)$ Zeitslots hatte jeder Knoten mit *hoher* Wahrscheinlichkeit ($> 1 - 1/n$) Erfolg.

Beweisskizze (vereinfacht):

- Schritt 1: Lokaler Erfolg - Mit konstanter Wahrscheinlichkeit (WS') sendet kein Knoten in der "Nähe"



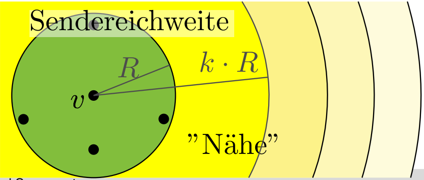
Local Broadcasting Algorithmus

Satz

Nach $O(\Delta \log n)$ Zeitslots hatte jeder Knoten mit *hoher* Wahrscheinlichkeit ($> 1 - 1/n$) Erfolg.

Beweisskizze (vereinfacht):

- Schritt 1: Lokaler Erfolg - Mit konstanter Wahrscheinlichkeit (WS') sendet kein Knoten in der "Nähe"
- Schritt 2: Globaler Erfolg - Interferenz von außerhalb ist mit konstanter Wahrscheinlichkeit "klein"



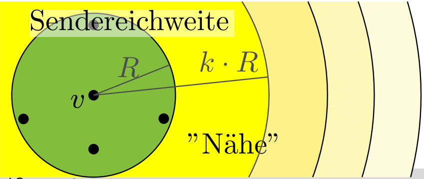
Local Broadcasting Algorithmus

Satz

Nach $O(\Delta \log n)$ Zeitslots hatte jeder Knoten mit *hoher* Wahrscheinlichkeit ($> 1 - 1/n$) Erfolg.

Beweisskizze (vereinfacht):

- Schritt 1: Lokaler Erfolg - Mit konstanter Wahrscheinlichkeit (WS') sendet kein Knoten in der "Nähe"
- Schritt 2: Globaler Erfolg - Interferenz von außerhalb ist mit konstanter Wahrscheinlichkeit "klein"
- Insgesamt: Je $O(\Delta)$ Zeitslots: Konstante Erfolgswahrscheinlichkeit



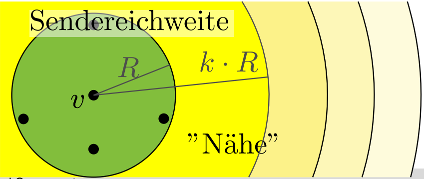
Local Broadcasting Algorithmus

Satz

Nach $O(\Delta \log n)$ Zeitslots hatte jeder Knoten mit *hoher* Wahrscheinlichkeit ($> 1 - 1/n$) Erfolg.

Beweisskizze (vereinfacht):

- Schritt 1: Lokaler Erfolg - Mit konstanter Wahrscheinlichkeit (WS') sendet kein Knoten in der “Nähe”
- Schritt 2: Globaler Erfolg - Interferenz von außerhalb ist mit konstanter Wahrscheinlichkeit “klein”
- Insgesamt: Je $O(\Delta)$ Zeitslots: Konstante Erfolgswahrscheinlichkeit
- $O(\Delta \log n)$ Zeitslots: Erfolg mit hoher Wahrscheinlichkeit



Local Broadcasting

Local Broadcasting Varianten (ohne Beweis)

1. Nach $O(\Delta \log n)$ Zeitslots hatte jeder Knoten mit hoher Wahrscheinlichkeit (WS') Erfolg.

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Local Broadcasting

Local Broadcasting Varianten (ohne Beweis)

1. Nach $O(\Delta \log n)$ Zeitslots hatte jeder Knoten mit hoher Wahrscheinlichkeit (WS') Erfolg.
2. Nach $O(\Delta)$ Zeitslots hatte jeder Knoten mit konstanter Wahrscheinlichkeit (WS') Erfolg.

Local Broadcasting

Karlsruher Institut für Technologie

Local Broadcasting Varianten (ohne Beweis)

1. Nach $O(\Delta \log n)$ Zeitslots hatte jeder Knoten mit hoher Wahrscheinlichkeit (WS') Erfolg.
2. Nach $O(\Delta)$ Zeitslots hatte jeder Knoten mit konstanter Wahrscheinlichkeit (WS') Erfolg.
3. Eine unabhängige Menge an Knoten hat (mittels erhöhter Sendewahrscheinlichkeit) nach $O(\log n)$ Zeitslots mit hoher Wahrscheinlichkeit (WS') Erfolg.

Erinnerung: TDMA - koordinierter Medienzugriff

- Übertragungen bekommen Slots in Zeitraster zugewiesen
  - *Time Division Multiple Access (TDMA)*
- Übertragungen sind *gleichzeitig* möglich, wenn sie weit genug auseinander liegen

Erinnerung: TDMA - koordinierter Medienzugriff

- Übertragungen bekommen Slots in Zeitraster zugewiesen
  - *Time Division Multiple Access (TDMA)*
- Übertragungen sind *gleichzeitig* möglich, wenn sie weit genug auseinander liegen

Erinnerung: TDMA - koordinierter Medienzugriff

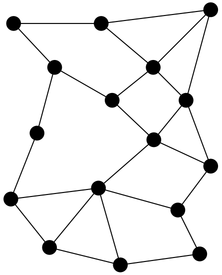
- Übertragungen bekommen Slots in Zeitraster zugewiesen
  - *Time Division Multiple Access (TDMA)*
- Übertragungen sind *gleichzeitig* möglich, wenn sie weit genug auseinander liegen

Jetzt:

- Knotenfärbung soll TDMA-Schedule für Local Broadcasts berechnen
- Im geometrischen SINR Modell ist k-Distanz Färbung notwendig (k-Hop Färbung nicht ausreichend)

Berechnung Local Broadcasting Schedule

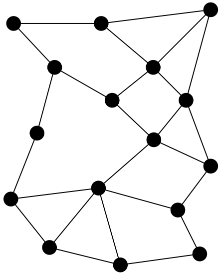
Ablauf



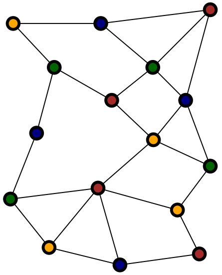
Berechnung Local Broadcasting Schedule

Ablauf

1. Berechnung einer 1-Hop Knotenfärbung
   - 2 Algorithmen: RandColor und ColorReduction



$\rightarrow$

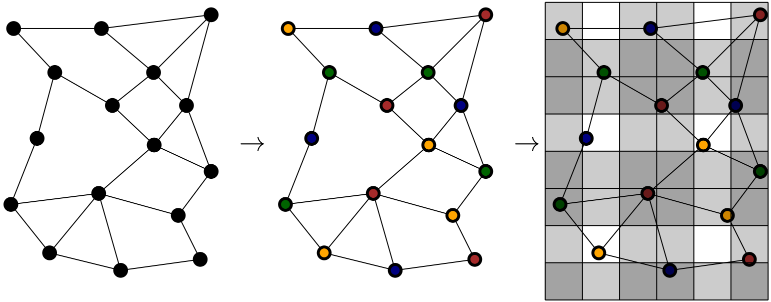


Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Berechnung Local Broadcasting Schedule

Ablauf

1. Berechnung einer 1-Hop Knotenfärbung
   - 2 Algorithmen: RandColor und ColorReduction
2. Techniken um die valid Färbung auf Distanz-k auszuweiten
   - 3 Techniken: Sendeleistung, Position, Steinerknoten



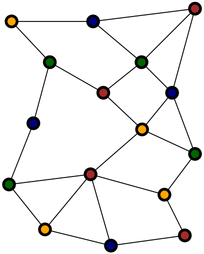
Erinnerung: Knotenfärbung

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 $|c| = \max_{v \in V} c(v)$

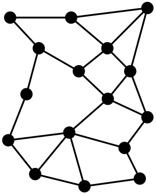


Färbung 1: RandColor

Karlsruher Institut für Technologie

RandColor Algorithmus

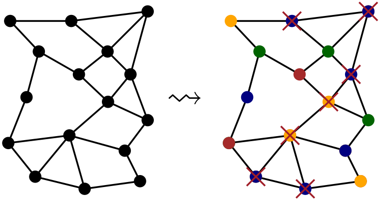
1. Wähle zufällig eine Farbe aus $\{1, \ldots, 4\Delta\}$
2. In jeder Runde
   - Wird ein Konflikt festgestellt, wähle neue zufällige Farbe aus $\{1, \ldots, 4\Delta\}$



Färbung 1: RandColor

RandColor Algorithmus

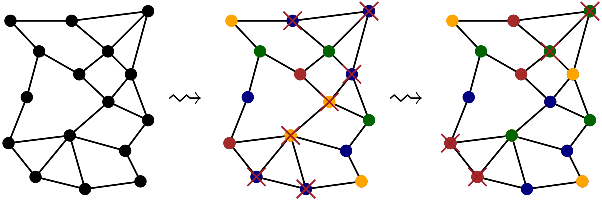
1. Wähle zufällig eine Farbe aus $\{1, \ldots, 4\Delta\}$
2. In jeder Runde
   - Wird ein Konflikt festgestellt, wähle neue zufällige Farbe aus $\{1, \ldots, 4\Delta\}$



Färbung 1: RandColor

RandColor Algorithmus

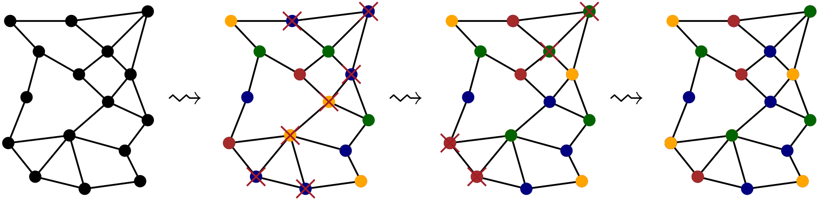
1. Wähle zufällig eine Farbe aus $\{1, \ldots, 4\Delta\}$
2. In jeder Runde
   - Wird ein Konflikt festgestellt, wähle neue zufällige Farbe aus $\{1, \ldots, 4\Delta\}$



Färbung 1: RandColor

RandColor Algorithmus

1. Wähle zufällig eine Farbe aus $\{1, \ldots, 4\Delta\}$
2. In jeder Runde
   - Wird ein Konflikt festgestellt, wähle neue zufällige Farbe aus $\{1, \ldots, 4\Delta\}$

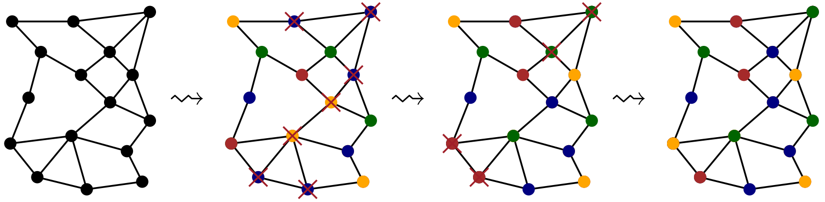


Färbung 1: RandColor

Karlsruher Institut für Technologie

RandColor Algorithmus

1. Wähle zufällig eine Farbe aus $\{1, \ldots, 4\Delta\}$
2. In jeder Runde
   - Wird ein Konflikt festgestellt, wähle neue zufällige Farbe aus $\{1, \ldots, 4\Delta\}$



Satz A

Ohne Interferenz: RandColor berechnet in $O(\log n)$ eine 4Δ-Knotenfärbung

Färbung 1: RandColor

Satz B

RandColor berechnet eine 4Δ-Knotenfärbung in $O(\Delta \log n)$ Zeitslots im SINR Modell

Beweisidee:

- Wird Local Broadcasting für die Runden genutzt, folgt aus Satz A direkt eine Laufzeit von $O(\Delta \log^2 n)$ (klar?)

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Färbung 1: RandColor

Satz B

RandColor berechnet eine 4Δ-Knotenfärbung in $O(\Delta \log n)$ Zeitslots im SINR Modell

Beweisidee:

- Wird Local Broadcasting für die Runden genutzt, folgt aus Satz A direkt eine Laufzeit von $O(\Delta \log^2 n)$ (klar?)
- Wir nutzen Local Broadcasting Variante 2 für jede Runde
  - Local Broadcasting (V2): Local Broadcast in $O(\Delta)$ Zeit mit konstanter Erfolgswahrscheinlichkeit

Färbung 1: RandColor

Satz B

RandColor berechnet eine 4Δ-Knotenfärbung in $O(\Delta \log n)$ Zeitslots im SINR Modell

Beweisidee:

- Wird Local Broadcasting für die Runden genutzt, folgt aus Satz A direkt eine Laufzeit von $O(\Delta \log^2 n)$ (klar?)
- Wir nutzen Local Broadcasting Variante 2 für jede Runde
  - Local Broadcasting (V2): Local Broadcast in $O(\Delta)$ Zeit mit konstanter Erfolgswahrscheinlichkeit
- Wahrscheinlichkeiten im probabilistischen Algorithmus und der Kommunikation können gemeinsam betrachtet werden

Färbung 1: RandColor

Satz B

RandColor berechnet eine 4Δ-Knotenfärbung in $O(\Delta \log n)$ Zeitslots im SINR Modell

Beweisidee:

- Wird Local Broadcasting für die Runden genutzt, folgt aus Satz A direkt eine Laufzeit von $O(\Delta \log^2 n)$ (klar?)
- Wir nutzen Local Broadcasting Variante 2 für jede Runde
  - Local Broadcasting (V2): Local Broadcast in $O(\Delta)$ Zeit mit konstanter Erfolgswahrscheinlichkeit
- Wahrscheinlichkeiten im probabilistischen Algorithmus und der Kommunikation können gemeinsam betrachtet werden
- Erneut: Konstante "Erfolgswahrscheinlichkeit" pro Runde (je $O(\Delta)$ Slots)
- Insgesamt: $O(\Delta \log n)$ Zeitslots

Färbung 2: ColorReduction

Karlsruher Institut für Technologie

(Triviale) ColorReduction

1. Beginne mit $d = O(\Delta)$ Farben
2. In Runde i, die Knoten mit Farbe i sind *aktiv*
   - wählen eine neue Farbe aus $\{1, \ldots, \Delta + 1\}$
3. Valide Färbung nach $d = O(\Delta)$ Runden

Färbung 2: ColorReduction

(Triviale) ColorReduction

1. Beginne mit $d = O(\Delta)$ Farben
2. In Runde i, die Knoten mit Farbe i sind *aktiv*
   - wählen eine neue Farbe aus $\{1, \dots, \Delta + 1\}$
3. Valide Färbung nach $d = O(\Delta)$ Runden

Satz

Der triviale ColorReduction Algorithm berechnet eine $(\Delta + 1)$-Färbung in $O(\Delta \log n)$ Zeitslots bei synchronem Start

Färbung 2: ColorReduction

(Triviale) ColorReduction

1. Beginne mit $d = O(\Delta)$ Farben
2. In Runde i, die Knoten mit Farbe i sind *aktiv*
   - wählen eine neue Farbe aus $\{1, \ldots, \Delta + 1\}$
3. Valide Färbung nach $d = O(\Delta)$ Runden

Satz

Der triviale ColorReduction Algorithm berechnet eine $(\Delta + 1)$-Färbung in $O(\Delta \log n)$ Zeitslots bei synchronem Start

Beweisidee:

- Aktive Knoten sind unabhängig $\Rightarrow$ $O(\log n)$ je Runde mittels Local Broadcast, Variante 3

Färbung 2: ColorReduction

(Triviale) ColorReduction

1. Beginne mit $d = O(\Delta)$ Farben
2. In Runde i, die Knoten mit Farbe i sind *aktiv*
   - wählen eine neue Farbe aus $\{1, \ldots, \Delta + 1\}$
3. Valide Färbung nach $d = O(\Delta)$ Runden

Satz

Der triviale ColorReduction Algorithm berechnet eine $(\Delta + 1)$-Färbung in $O(\Delta \log n)$ Zeitslots bei synchronem Start

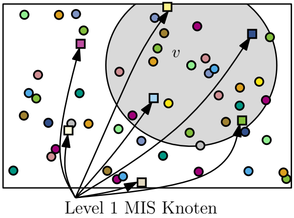
Beweisidee:

- Aktive Knoten sind unabhängig $\Rightarrow$ $O(\log n)$ je Runde mittels Local Broadcast, Variante 3

Geht das auch ohne synchronen Start?

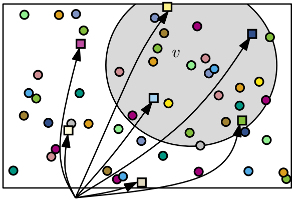
Färbung 2: ColorReduction

Idee: 2-schichtiges MIS zur Koordination der aktiven Farbe(n)

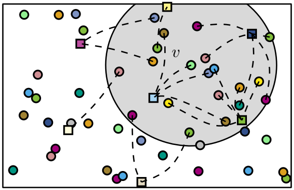


Färbung 2: ColorReduction

Idee: 2-schichtiges MIS zur Koordination der aktiven Farbe(n)



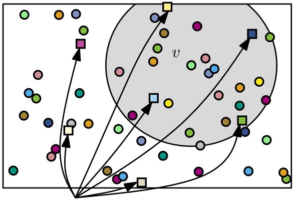
Level 1 MIS Knoten



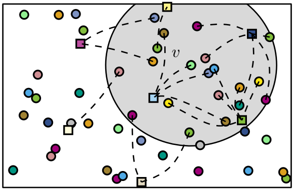
Restliche Knoten wählen MIS Knoten
und beantragen Aktivitätsintervall

Färbung 2: ColorReduction

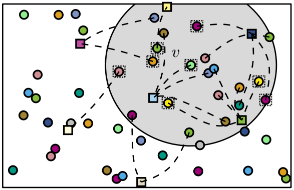
Idee: 2-schichtiges MIS zur Koordination der aktiven Farbe(n)



Level 1 MIS Knoten



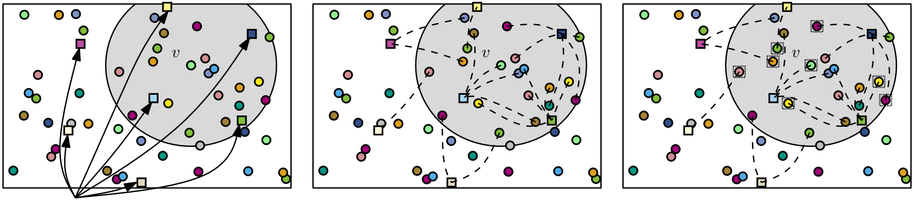
Restliche Knoten wählen MIS Knoten und beantragen Aktivitätsintervall



MIS Knoten bestimmen aktive Farbe. Aktive Knoten nehmen an (schnellem) Level 2 MIS teil

Färbung 2: ColorReduction

Idee: 2-schichtiges MIS zur Koordination der aktiven Farbe(n)



Level 1 MIS Knoten

Restliche Knoten wählen MIS Knoten und beantragen Aktivitätsintervall

MIS Knoten bestimmen aktive Farbe. Aktive Knoten nehmen an (schnellem) Level 2 MIS teil

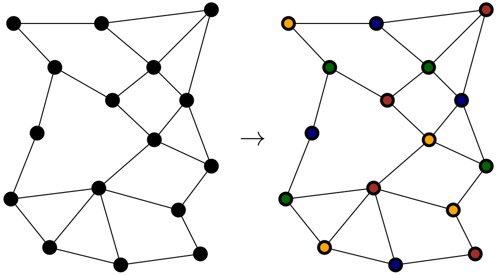
Satz (ohne Beweis)

Der ColorReduction Algorithm berechnet eine $(\Delta + 1)$-Färbung in $O(\Delta \log n)$ Zeitslots bei asynchronem Start

Zwischenstand: Local Broadcasting Schedule

Ablauf

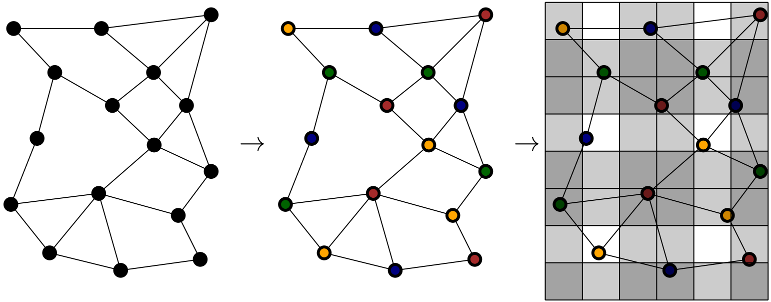
1. Berechnung einer 1-Hop Knotenfärbung
   - 2 Algorithmen: RandColor und ColorReduction



Zwischenstand: Local Broadcasting Schedule

Ablauf

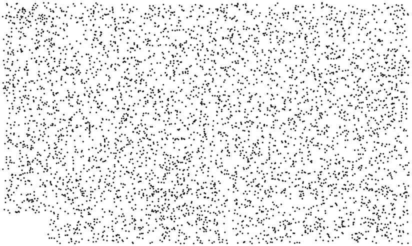
1. Berechnung einer 1-Hop Knotenfärbung
   - 2 Algorithmen: RandColor und ColorReduction
2. Techniken um die valid Färbung auf Distanz-k auszuweiten
   - 3 Techniken: Sendeleistung, Position, Steinerknoten



Local Broadcasting Schedule

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann mittels Knotenfärbung und Knotenpositionen berechnet werden.

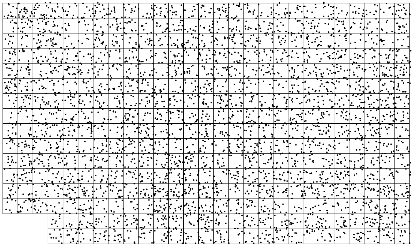


Local Broadcasting Schedule

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann mittels Knotenfärbung und Knotenpositionen berechnet werden.

- Knotenposition erlaubt Einteilung in Grid



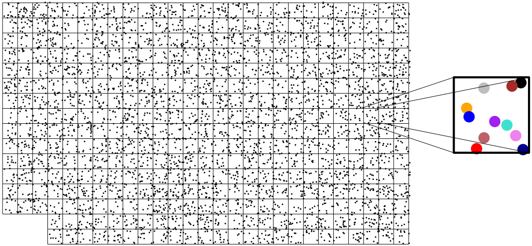
Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Local Broadcasting Schedule

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann mittels Knotenfärbung und Knotenpositionen berechnet werden.

- Knotenposition erlaubt Einteilung in Grid
- Pro Zelle nur ein Knoten je Farbe



Institut für Theoretische Informatik
Lehrstuhl Algorithmik

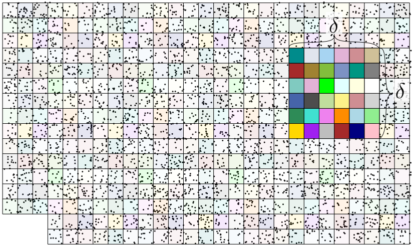
Local Broadcasting Schedule

KIT
Karlsruher Institut für Technologie

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann mittels Knotenfärbung und Knotenpositionen berechnet werden.

- Knotenposition erlaubt Einteilung in Grid
- Pro Zelle nur ein Knoten je Farbe
- Basierend auf Knotenposition kann eine weitere Färbung berechnet werden

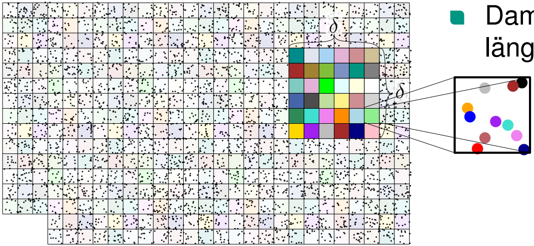


Local Broadcasting Schedule

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann mittels Knotenfärbung und Knotenpositionen berechnet werden.

- Knotenposition erlaubt Einteilung in Grid
- Pro Zelle nur ein Knoten je Farbe
- Basierend auf Knotenposition kann eine weitere Färbung berechnet werden



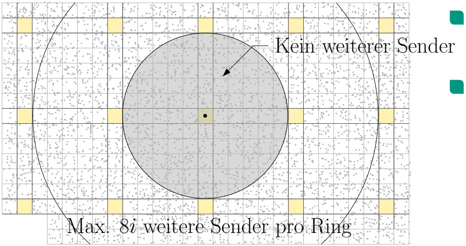
- Damit ergibt sich um $\delta^2 \in O(1)$ längerer Schedule

Local Broadcasting Schedule

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann mittels Knotenfärbung und Knotenpositionen berechnet werden.

- Knotenposition erlaubt Einteilung in Grid
- Pro Zelle nur ein Knoten je Farbe
- Basierend auf Knotenposition kann eine weitere Färbung berechnet werden



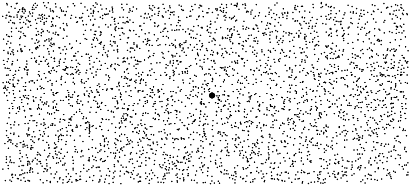
- Damit ergibt sich um $\delta^2 \in O(1)$ längerer Schedule
- In diesem Schedule ist jede Übertragung ohne Kollision
  - Keine Knoten in der Nähe senden
  - Die Interferenz der gleichzeitig sendenden Knoten ist *klein genug*

Local Broadcasting Schedule

Weitere Techniken mit ähnlicher Idee:

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch Erhöhung der Sendeleistung während der Knotenfärbung berechnet werden.

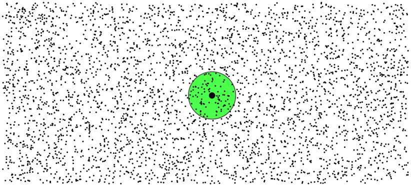


Local Broadcasting Schedule

Weitere Techniken mit ähnlicher Idee:

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch Erhöhung der Sendeleistung während der Knotenfärbung berechnet werden.

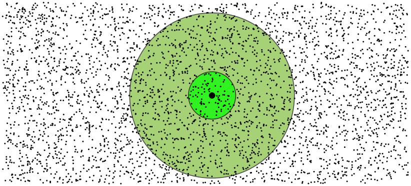


Local Broadcasting Schedule

Weitere Techniken mit ähnlicher Idee:

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch Erhöhung der Sendeleistung während der Knotenfärbung berechnet werden.



- Durch Erhöhung der Sendereichweite ist Berechnung der Distanz-k Färbung direkt möglich

Local Broadcasting Schedule

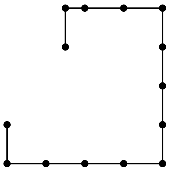
Weitere Techniken mit ähnlicher Idee:

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch Erhöhung der Sendeleistung während der Knotenfärbung berechnet werden.

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch mittels zusätzlichem Einfügen von $O(k)$ Steinerknoten pro Knoten und einer $k - Hop$ Knotenfärbung erreicht werden.



Local Broadcasting Schedule

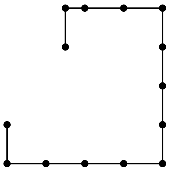
Weitere Techniken mit ähnlicher Idee:

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch Erhöhung der Sendeleistung während der Knotenfärbung berechnet werden.

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch mittels zusätzlichem Einfügen von $O(k)$ Steinerknoten pro Knoten und einer $k - Hop$ Knotenfärbung erreicht werden.



- Problem: k-Hop Knotenfärbung erreicht nicht alle Knoten in Distanz $k \cdot R$

Local Broadcasting Schedule

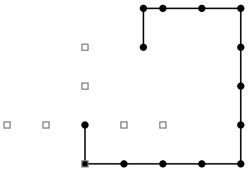
Weitere Techniken mit ähnlicher Idee:

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch Erhöhung der Sendeleistung während der Knotenfärbung berechnet werden.

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch mittels zusätzlichem Einfügen von $O(k)$ Steinerknoten pro Knoten und einer $k - Hop$ Knotenfärbung erreicht werden.



- Problem: k-Hop Knotenfärbung erreicht nicht alle Knoten in Distanz $k \cdot R$
- Füge $O(nk)$ Steinerknoten hinzu

Local Broadcasting Schedule

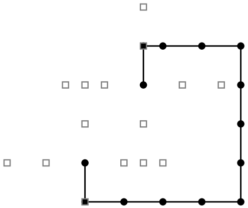
Weitere Techniken mit ähnlicher Idee:

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch Erhöhung der Sendeleistung während der Knotenfärbung berechnet werden.

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch mittels zusätzlichem Einfügen von $O(k)$ Steinerknoten pro Knoten und einer $k - Hop$ Knotenfärbung erreicht werden.



- Problem: k-Hop Knotenfärbung erreicht nicht alle Knoten in Distanz $k \cdot R$
- Füge $O(nk)$ Steinerknoten hinzu

Local Broadcasting Schedule

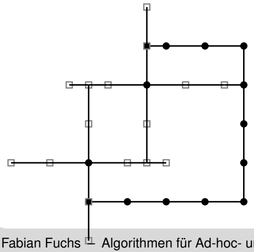
Weitere Techniken mit ähnlicher Idee:

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch Erhöhung der Sendeleistung während der Knotenfärbung berechnet werden.

Satz

Ein kollisionsfreier Local Broadcasting Schedule kann durch mittels zusätzlichem Einfügen von $O(k)$ Steinerknoten pro Knoten und einer $k - Hop$ Knotenfärbung erreicht werden.



(Eigentlich: Steinerknoten für jeden Knoten)

- Problem: k-Hop Knotenfärbung erreicht nicht alle Knoten in Distanz $k \cdot R$
- Füge $O(nk)$ Steinerknoten hinzu
- Jetzt erreicht k-Hop Knotenfärbung alle nötigen Knoten

Institut für Theoretische Informatik
Lehrstuhl Algorithmik

Zum Mitnehmen

- Kommunikation im SINR Modell
  - Standard Local Broadcasting & Varianten
- Koordinierter Medienzugriff
  - Local Broadcasting Schedule bzw. TDMA-Schedule
  - Knotenfärbung: RandColor und ColorReduction im SINR Modell
  - Kombiniert mit Positionsinformation, Sendeleistungs-tuning oder zusätzlichen Steinerknoten: kollisionsfreier Local Broadcasting Schedule

Übung

Karlsruher Institut für Technologie

Jetzt: Übung

- Local Broadcasting
- RandColor

Local Broadcasting

Local Broadcasting

Jeder Knoten kennt n und Knotengrad Δ

1. Sende in jedem Zeitslot mit Wahrscheinlichkeit $1/O(\Delta)$

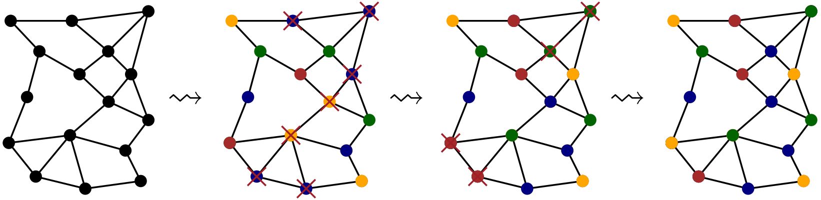
Fragestellungen:

- Wie groß sollte Sendewahrscheinlichkeit wirklich sein?
  - $1/O(\Delta) = \frac{1}{c\Delta}$ für welches c?
- Wie lange dauert es bis alle Knoten erfolgreich waren?

RandColor

RandColor Algorithmus

1. Verwende Sendewahrscheinlichkeit von Local Broadcasting
2. Wähle zufällig eine Farbe aus $\{1, \ldots, 4\Delta\}$
3. In jeder Runde
   - Wird ein Konflikt festgestellt, wähle neue zufällige Farbe aus $\{1, \ldots, 4\Delta\}$



Fragestellungen:

- Wie lange sollte eine Runde sein?
- Wie lange dauert es bis alle Knoten erfolgreich waren?

Literatur

1. O. Goussevskaia, T. Moscibroda, R. Wattenhofer: *Local broadcasting in the physical interference model*. In: *DIALM-POMC'08*
2. C. Avin, Y. Emek, E. Kantor, Z. Lotker, D. Peleg, L. Roditty : *SINR Diagrams: Towards Algorithmically Usable SINR Models of Wireless Networks*, In: *PODC'09*
3. F. Fuchs, R. Prutkin: *Simple Distributed* $\Delta + 1$ *Coloring in the SINR Model*. In: *SIROCCO'15*
4. F. Fuchs, D. Wagner: *On Local Broadcasting Schedules and CONGEST Algorithms in the SINR Modell*. In: *ALGOSENSORS'13*
5. L. Barenboim: *Nearly Optimal Local Broadcasting in the SINR Model with Feedback*. In: *SIROCCO'15*
6. B. Derbel, E. Talbi: *Distributed Node Coloring in the SINR Model*. In: *ICDCS'10*