

# Algorithmen für Ad-hoc- und Sensornetze

Nachspielzeit zu VL 05

Dr. rer. nat. Bastian Katz

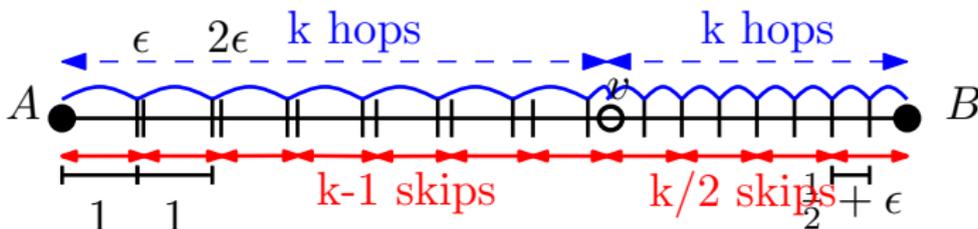
Lehrstuhl für Algorithmik I  
Institut für theoretische Informatik  
Universität Karlsruhe (TH)  
Karlsruher Institut für Technologie

3. Juni 2009

(Version 2 vom 5. Juni 2009)

## Erinnerung: UDG-Lokalisierung in 1D

- Wenn man nur Hops zu Anker kennt, ist es unmöglich, einen Fehler zu garantieren, der für jeden Knoten innerhalb eines konstanten Vielfachen des Fehlers eines Optimalen Algorithmus liegt.
  - Fehler: Abstand zwischen echter Knotenposition und Ergebnis der Lokalisierung
  - Optimaler Algo: Minimiert Fehler über alle möglichen Einbettungen
- Kennt man *hops* und *skips*, kann man (fast) optimal lokalisieren.



Bastian Katz – Algorithmen für Ad-hoc- und Sensornetze

# Skips (formaler)

## Definition

Ein Pfad  $A = v_0, u_1, v_1, \dots, u_s, v_s = v$  ist ein Skip-Pfad der Länge  $s$  zwischen  $A$  und  $v$ , wenn wenn

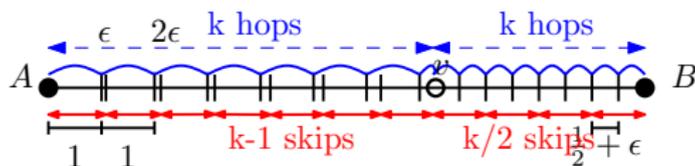
$$\gg d(A, v_i) < d(A, v_{i+1})$$

$$\gg d(v_i, v_{i+1}) > 1 \text{ (also } (v_i, v_{i+1}) \notin E \text{)}$$

Die Länge eines längsten solchen Pfades ist die Skip-Entfernung von  $v$  zu  $A$ .

$\gg$  hops sind obere Schranke an Entfernung zum Anker

$\gg$  skips sind untere Schranke



Wie berechnet man skips?

## Erinnerung: Hop-Berechnung

### Wie berechnet man hops?

- jeder Knoten  $v$  startet mit  $h_v = \infty$ .
- hört ein Knoten  $v$  von einem Nachbarn  $u$  mit  $h_u + 1 < h_v$ , verringert er  $h_v$  zu  $h_v = h_u + 1$ .
- wenn ein Knoten sein  $h_v$  verringert, teilt er das allen Nachbarn mit.
- Startschuss: Ankerknoten verringert seinen Abstand auf  $h_A = 0$ .

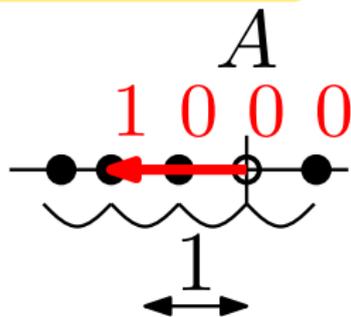
Wenn man hops zu mehreren Ankern berechnen will, muss der jeweilige Anker Teil der Nachricht sein!

*Ich,  $v$  habe meinen hop-Abstand zu Anker  $A$  auf  $h_v$  reduziert.*

## Wie berechnet man *skips*?

- jeder Knoten  $v$  startet mit  $s_v = -1$ .
- hört ein Knoten  $v$  von einem Nicht-Nachbarn  $u$  mit  $s_u + 1 > s_v$ , erhöht er  $s_v$  zu  $s_v = s_u + 1$ .
- wenn ein Knoten sein  $s_v$  erhöht, teilt er das allen Zwei-hop-Nachbarn mit.
- Startschuss: Jeder Nachbar  $v$  des Ankerknoten und  $A$  selbst erhöht seinen Abstand auf  $s_v = 0$ .

- Wie verhindert man „Aufschaukeln“?
- Beleg für Erhöhung sind nur Nicht-Nachbarn  $u$ , die echt zwischen  $v$  und Anker liegen
- Dann liegt ein Nachbar mit niedrigerem Hop-Abstand dazwischen!



## Wie berechnet man *skips*?

- jeder Knoten  $v$  startet mit  $s_v = -1$ .
- hört ein Knoten  $v$  von einem Nicht-Nachbarn  $u$  mit  $s_u + 1 > s_v$ , erhöht er  $s_v$  zu  $s_v = s_u + 1$ .
- wenn ein Knoten sein  $s_v$  erhöht, teilt er das allen Zwei-hop-Nachbarn mit.
- Startschuss: Jeder Nachbar  $v$  des Ankerknoten und  $A$  selbst erhöht seinen Abstand auf  $s_v = 0$ .

- Wie verhindert man „Aufschaukeln“?
- Beleg für Erhöhung sind nur Nicht-Nachbarn  $u$ , die echt zwischen  $v$  und Anker liegen
- Dann liegt ein Nachbar mit niedrigerem Hop-Abstand dazwischen!

