

Vorlesung Internet of Everything Wintersemester 2017/18

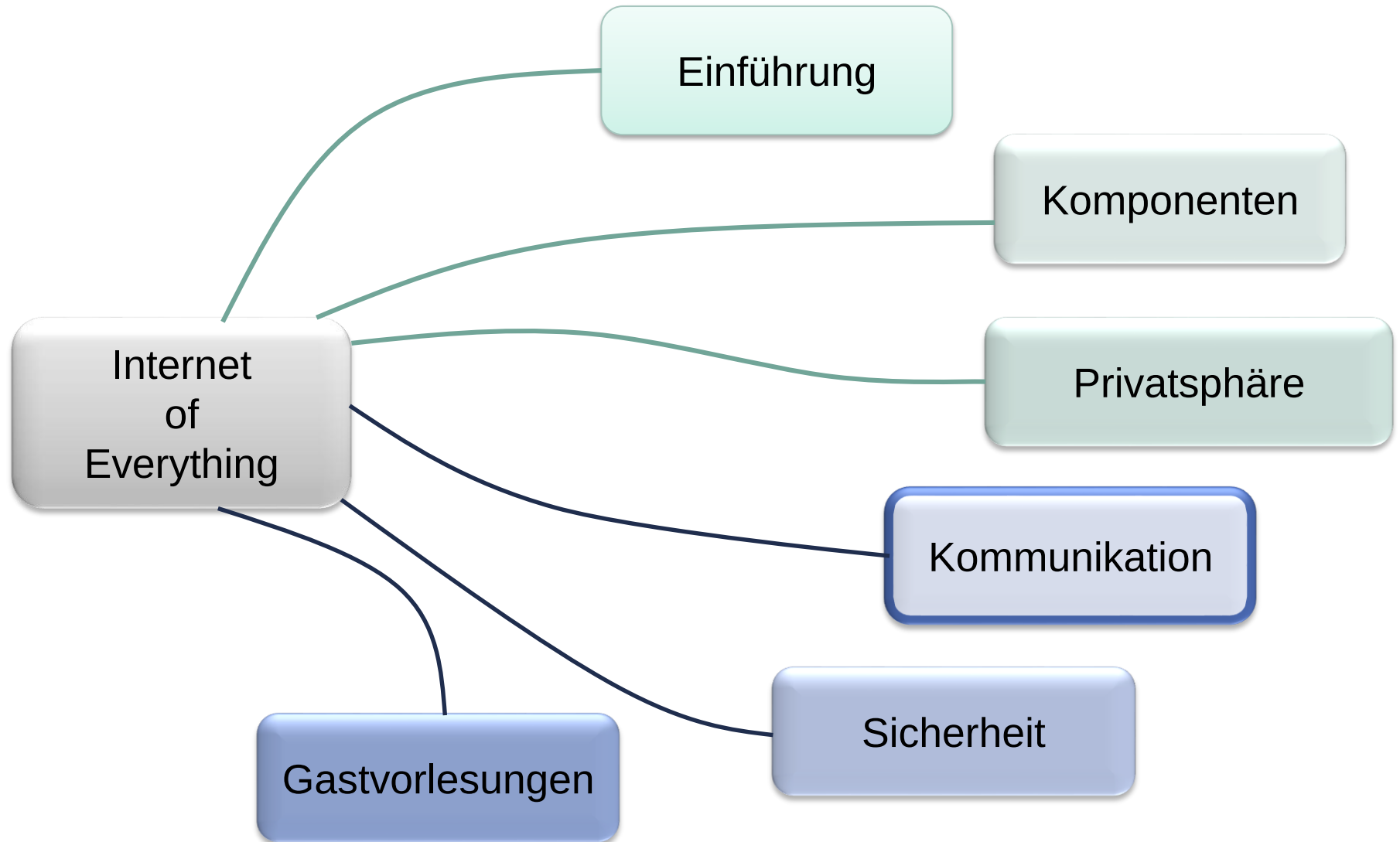
Kapitel 4.3 – Topologiekontrolle

Institut für Telematik, Prof. Zitterbart

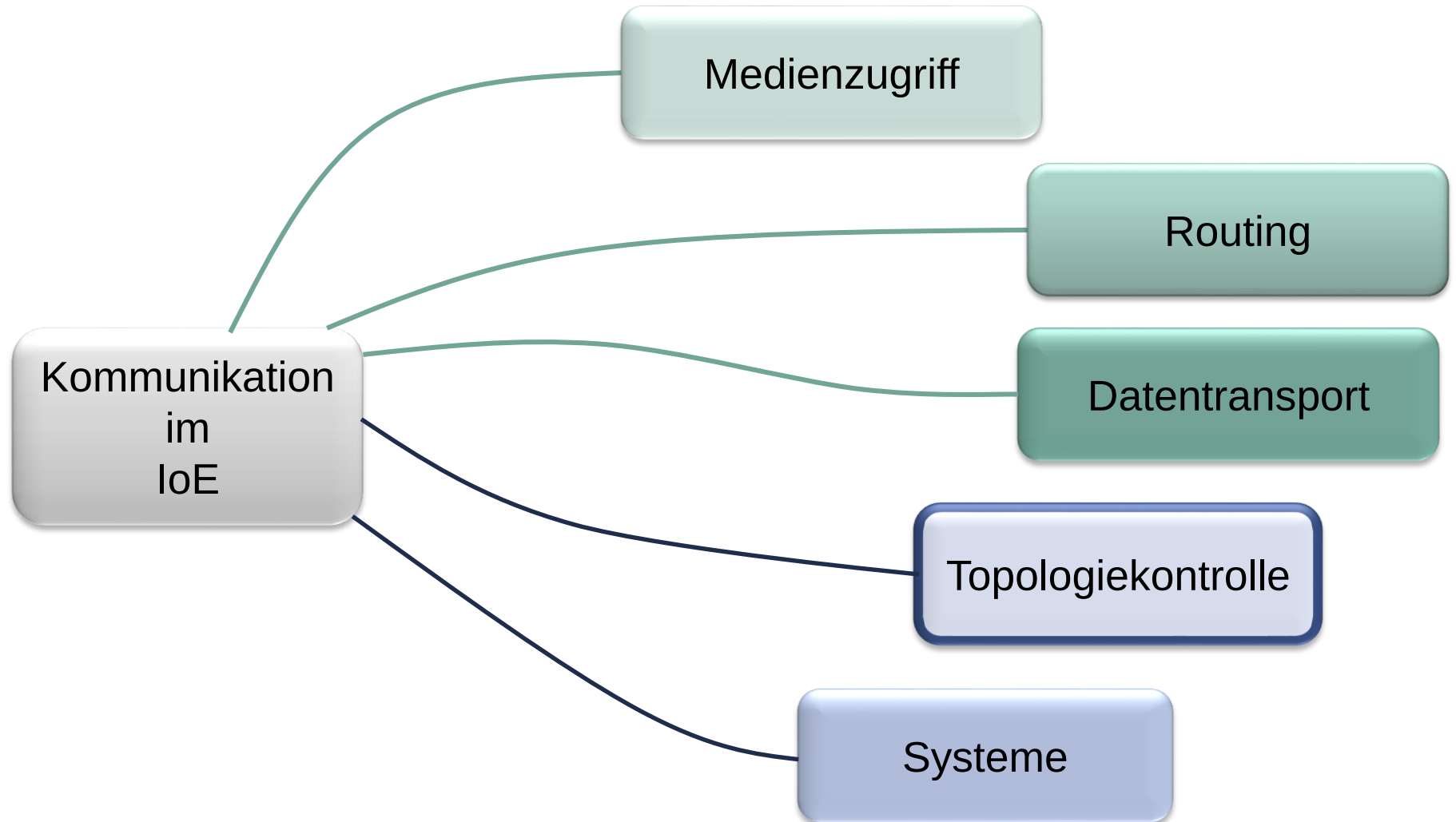


© Peter Baumung

Inhalte der Vorlesung



Kapitel 4.3: Topologiekontrolle



Einführung

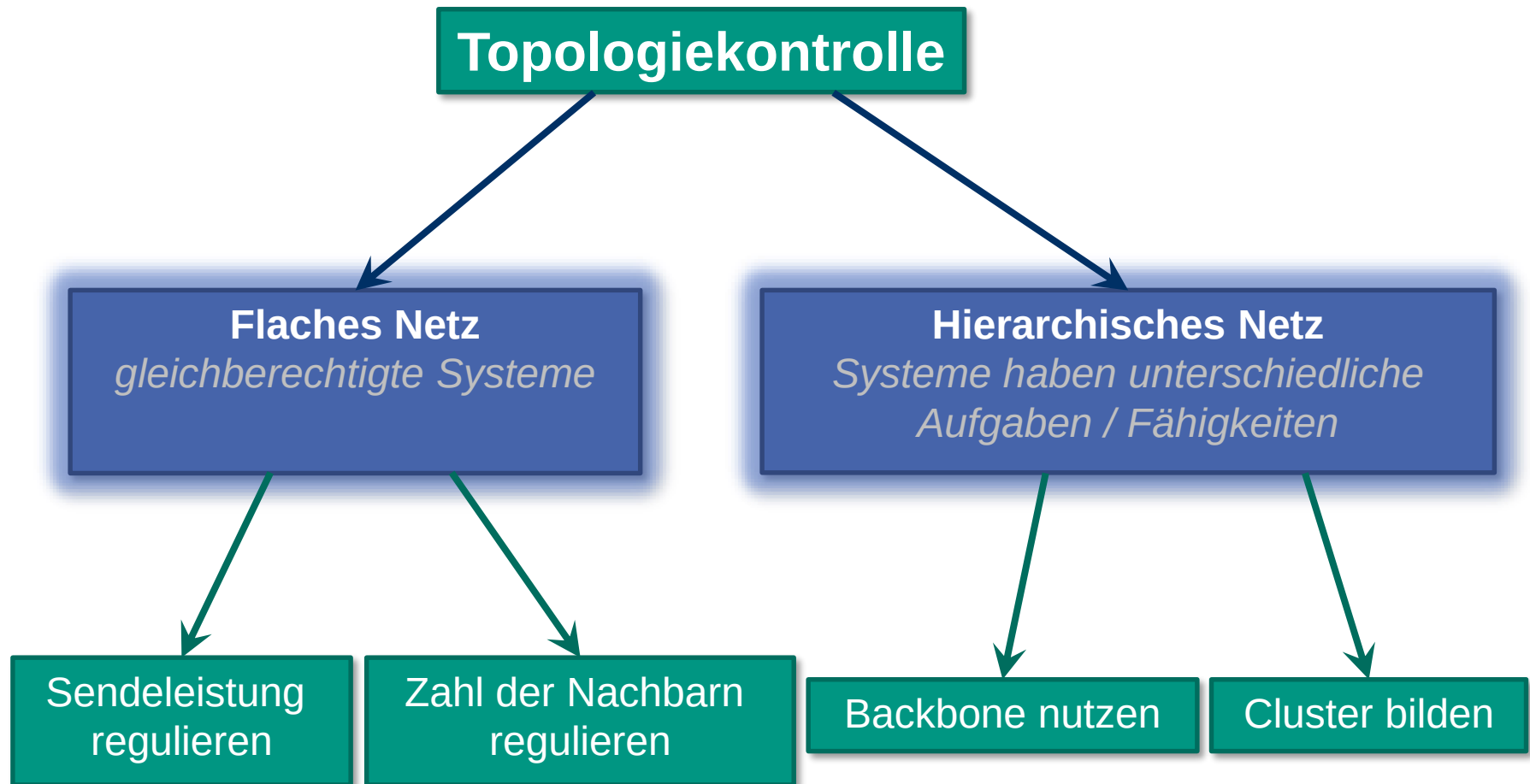
■ Topologie

- Aufbau von Verbindungen zwischen Systemen
 - Jede direkte Kommunikationsverbindung zwischen zwei Systemen ist Teil der Topologie
- Dichtes Netz führt zu komplexer Topologie
 - Maß der Dichte = Anzahl der Nachbarn
 - Abhängig von betrachteter **Schicht**

■ Idee

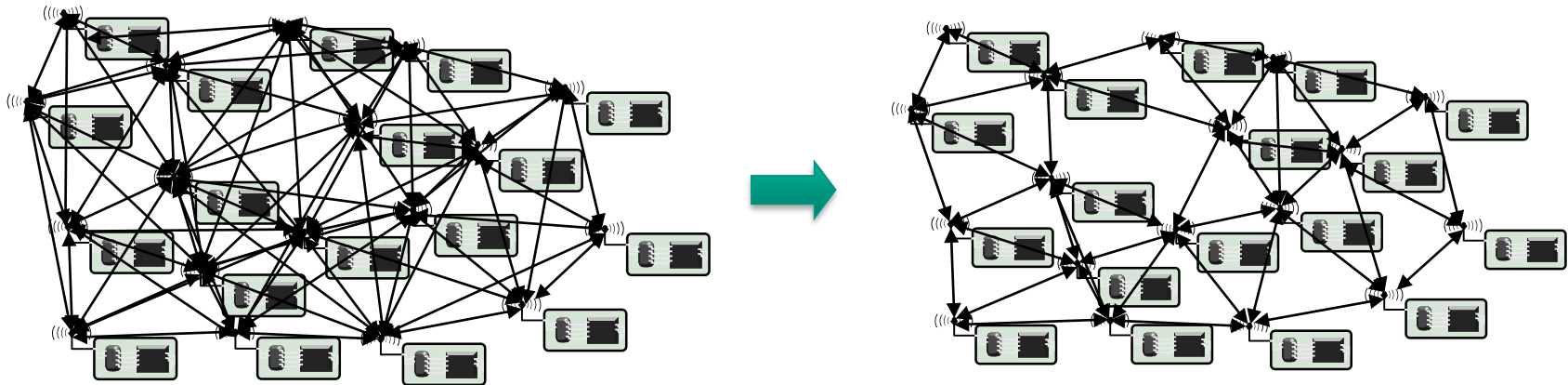
- Komplexität reduzieren
 - Durch Topologiekontrolle
 - Darf nicht zu restriktiv sein: Erhaltung der Konnektivität

Möglichkeiten der Kontrolle



Flache Netze

- Sendeleistung regeln
 - Nicht immer maximale Sendeleistung nutzen
 - Bezogen auf ganzes Netz oder einzelne Verbindungen
 - Weniger Nachbarn und Beeinflussungen
 - Einfachere Topologie, einfacheres Routing
 - Weniger Energieverbrauch des Sendevorgangs

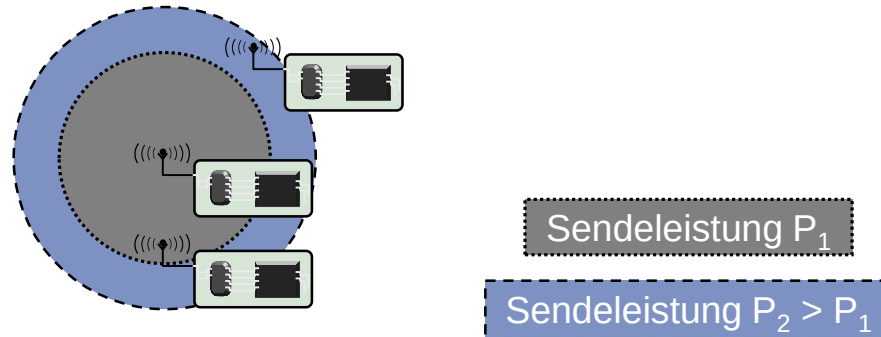


- Alternativ: Auf einzelne Verbindungen verzichten

Sendeleistung regulieren

■ Beobachtung

- Sendeleistung und Zahl der direkten Nachbarsysteme hängen unmittelbar zusammen



■ Problem

- „*Richtige*“ Sendeleistung finden
- Abhängig von genannten Vergleichskriterien
 - Geringe Sendeleistung spart Energie, verschlechtert aber Konnektivität, Durchsatz, etc.
 - Erwünscht sind geringe Sendeleistung, Konnektivität und hoher Durchsatz
 - Zweckdienlicher **Kompromiss** muss gefunden werden

Minimale maximale Sendeleistung (MMS)

■ Ziel

- Sendeleistung der einzelnen Systeme minimieren



[RaRo00]

■ Gegeben

- **Position** aller Systeme

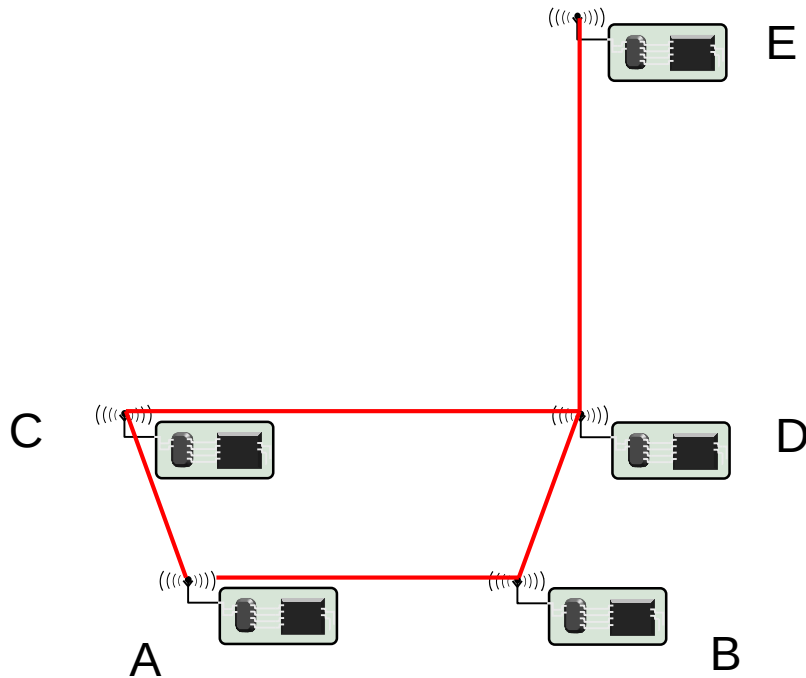
■ Zentralisierter Greedy-Algorithmus

- Für jedes System i , setze Sendeleistung $P_i=0$
- Zwei Teilnetze T_1, T_2 mit geringstem Abstand verbinden
- Sendeleistung P_i, P_j der zwei Systeme $i \in T_1, j \in T_2$, die am dichtesten zusammen liegen so erhöhen, dass diese miteinander kommunizieren können
- Abschließend Verbindungen, die Konnektivität nicht beeinflussen, entfernen

■ Bewertung

- Erhält die **Konnektivität** des Netzes
- **Zentralisierter** Ansatz ... in Sensornetzen so nicht praktikabel!

Beispiel MMS



■ 5 Teilgraphen

- A, B, C, D, E mit
- Sendeleistungen
 $P_A = P_B = P_C = P_D = P_E = 0$

■ (A, C) und (B, D) verbinden

- $P_A = P_C = 1$
- $P_B = P_D = 1$

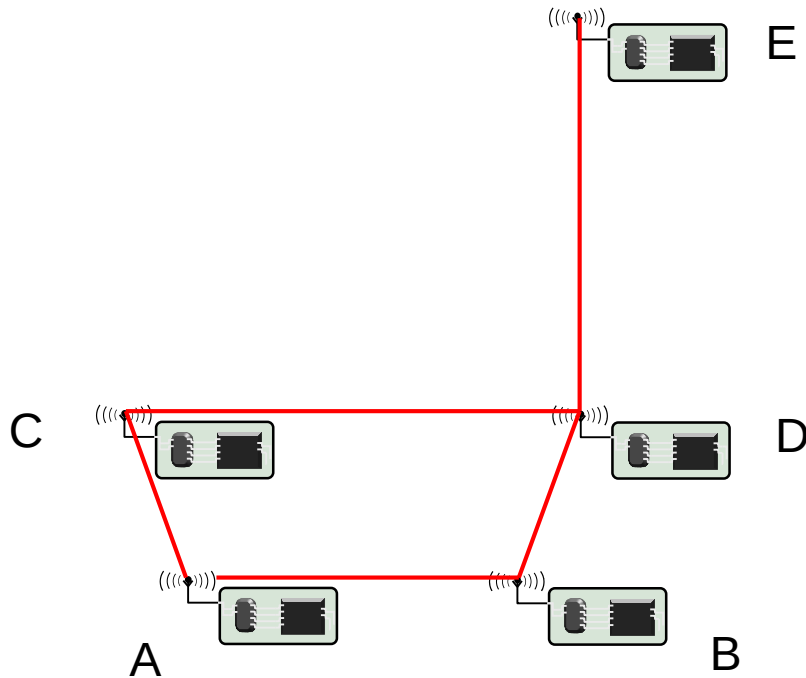
■ (AC, BD) verbinden

- $P_A = P_B = 2$

■ (ABCD, E) verbinden

- $P_D = P_E = 3$
- Seiteneffekt: Asymmetrische Verbindung DC

Beispiel MMS



■ Wie vereinfachen?

■ Möglichkeit 1

- $P_C=3$
- Verbindung CD hergestellt
- $P_A=P_B=1$
- Verbindung AB wird nicht mehr benötigt

■ Möglichkeit 2

- $P_C=1$ belassen
- Verbindung AB wird weiter benötigt

Hierarchische Netze (I)

■ Backbone-Netz

■ Systeme in zwei Klassen teilen

- Einige Systeme bilden Backbone
- Systeme außerhalb des Backbones dürfen nicht direkt kommunizieren
- Backbone bildet „dominating set“ D

(1) Jedes System ist entweder Teil von D

(2) Oder besitzt mindestens ein System aus D als direkten Nachbarn

- Backbone muss verbunden sein

■ Formal

- Netz als Graph $G = (V, E)$
- Gesucht ist Teilmenge D von V

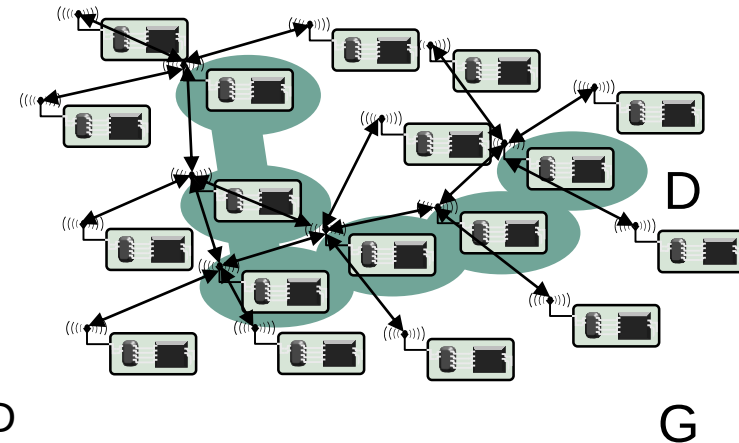
$$\forall v \in V : v \in D \vee \exists d \in D : (v, d) \in E$$



(1)



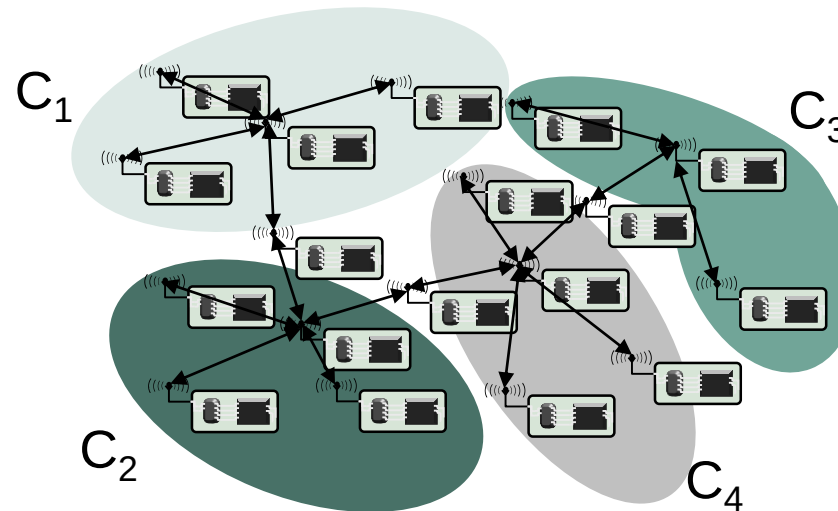
(2)



Hierarchische Netze (II)

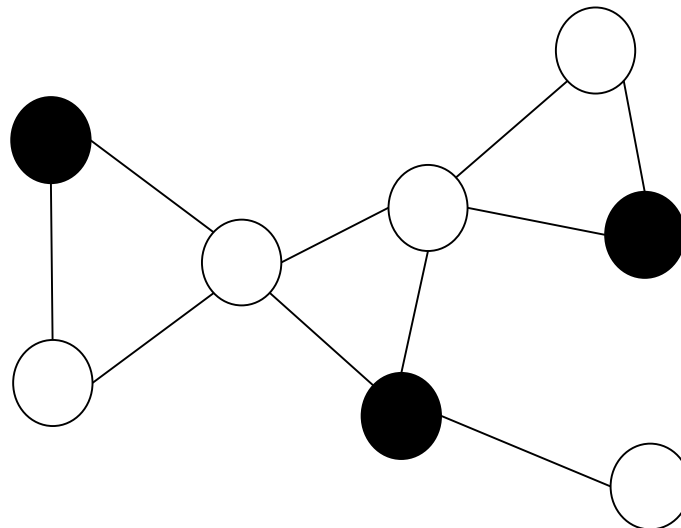
Cluster-Netze

- Sensoren in Gruppen (**Cluster**) C_x einteilen
 - Jedes System in genau einer Gruppe
 - Ausnahme: Systeme, die zwei oder mehr Gruppen verbinden



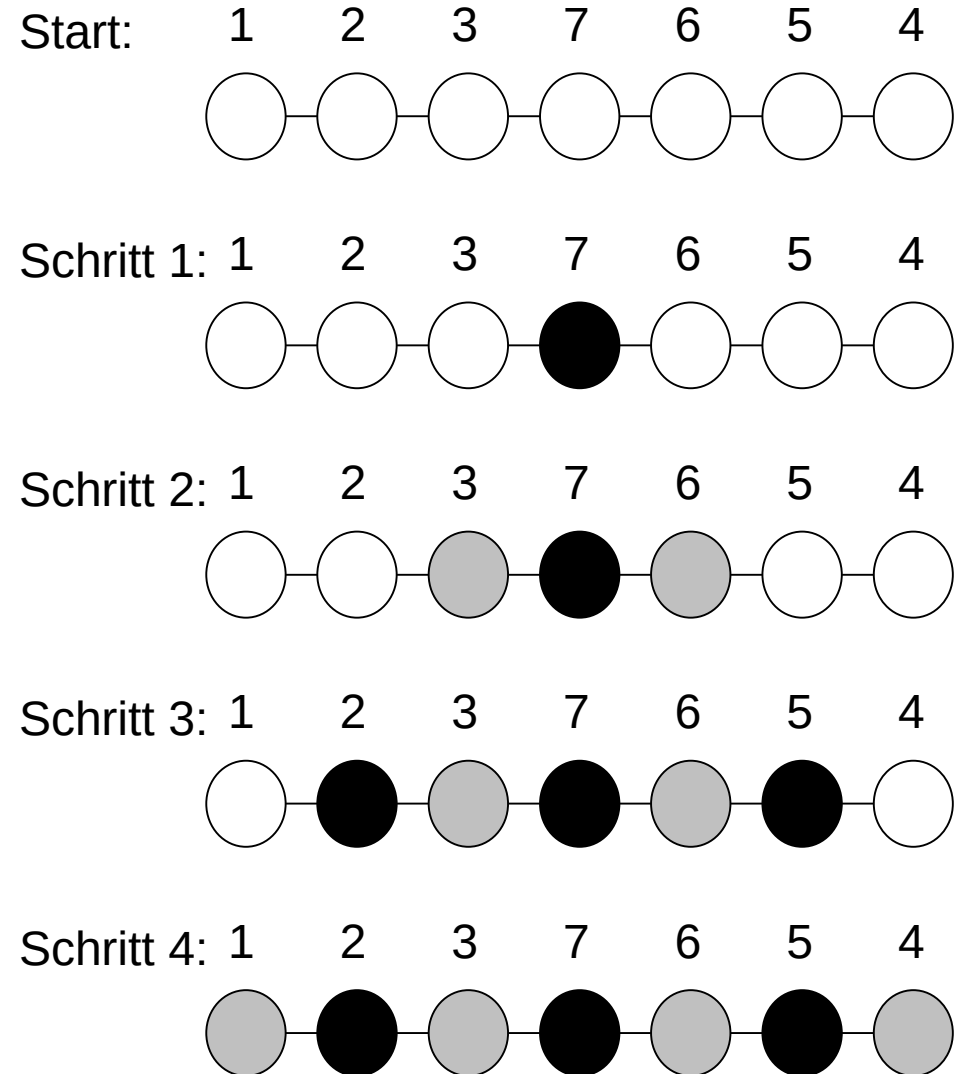
Clusterbildung

- Verschiedene Aspekte zu beachten
 - Vertreter der Gruppe (**Cluster-Head**)
 - Cluster-Heads bilden unabhängige Menge
 - Zwei Cluster-Heads dürfen nicht benachbart sein
 - Gesucht ist insbesondere maximale unabhängige Menge
 - Berechnung der maximalen unabhängigen Menge ist NP-vollständiges Problem
 - **Beispiel:** maximale unabhängige Menge



Maximale unabhängige Menge

- Gesucht ist Attribut um Knoten zu ordnen
 - Knoten-ID, Energiereserven, Bewegungsgeschwindigkeit, Kombinationen hiervon ...
- Knoten tauschen Attribut lokal aus
- Derjenige wird Cluster-Head, der größten Attributwert aufweist
- Einem Cluster-Head benachbarte Knoten werden von der Betrachtung ausgeschlossen



Bestimmung der Gateways

- Cluster-Heads gefunden (Menge C)
- Ziel: Cluster verbinden, Gateways finden

- Jeder Cluster-Head benötigt Verbindung zu allen anderen Cluster-Heads, die maximal 3 Schritte entfernt sind
 - Ergebnis: Backbone der Konnektivität sicherstellt

- Lässt sich formal auf Steinerbaum-Problem zurückführen
 - NP-vollständiges Problem
 - vereinfachter Fall, da C unabhängige Menge bildet

- Generelles Problem: Cluster-Heads werden mehr belastet, fallen schneller aus (Energie!)

LEACH

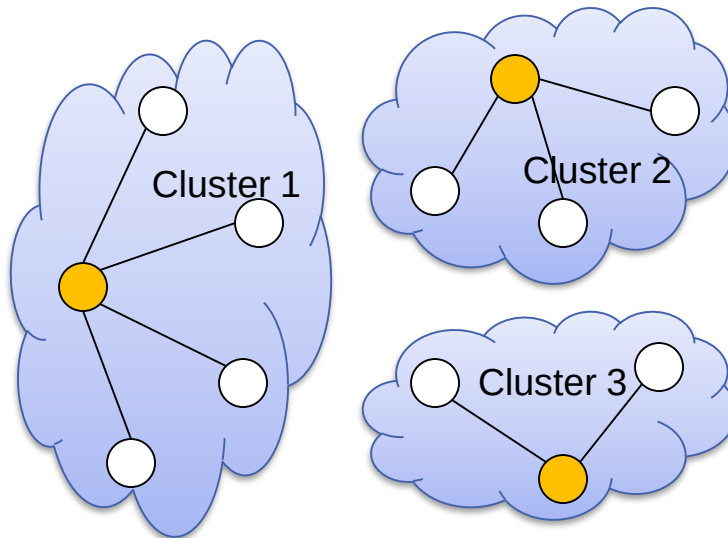
- Idee: Rotieren der Cluster-Heads sorgt für Verteilung der Last
 - Regelmäßige Neuverteilung der Aufgaben
 - Führt zu gleichmäßigerem Energiebedarf der Knoten
- Beispiel: „LEACH – Low-Energy Clustering Hierarchy“
 - Prinzip: $P\%$ der Knoten sind Cluster-Heads, Aufteilung wird regelmäßig geändert
- Merkmale von LEACH
 - Verteilter Aufbau einer Cluster-Topologie
 - Gleichmäßige Verteilung des Energieverbrauchs
 - Möglichkeit Daten am Cluster-Head zu aggregieren
- Annahme
 - Cluster-Heads kommunizieren (z.B. durch hohe Sendeleistung) direkt mit Basisstation



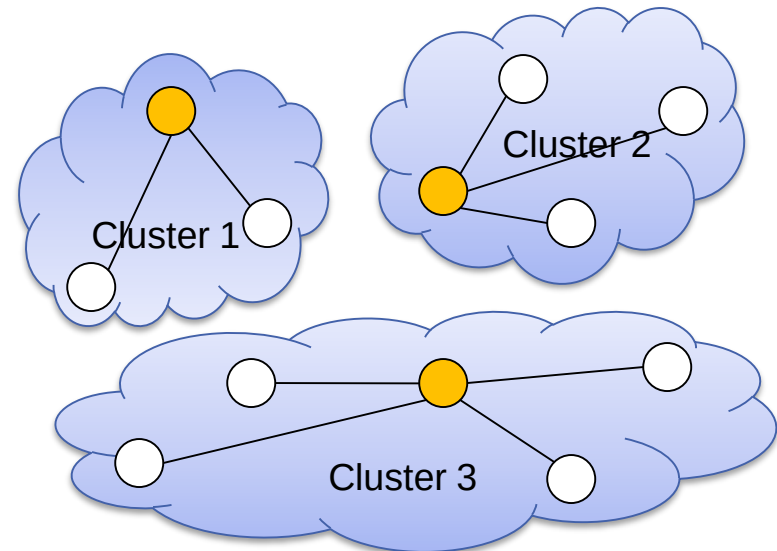
LEACH – Phasen

■ Vorgehen

- Knoten bestimmen sich selbständig mit bestimmter Wahrscheinlichkeit zu einem Cluster-Head und teilen dies den benachbarten Knoten mit
- Alle anderen Knoten ordnen sich dem nächsten in Reichweite befindlichen Cluster-Head zu



Clustering zum Zeitpunkt t_1



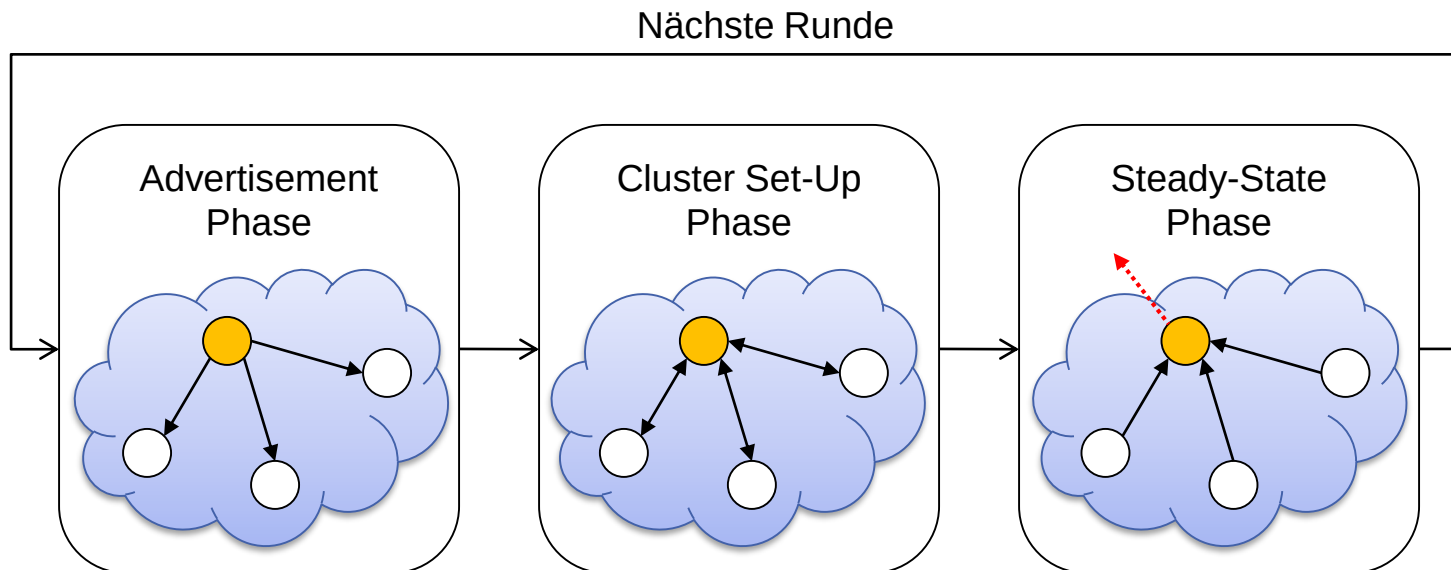
Clustering zum Zeitpunkt $t_1 + d$



LEACH

■ Ablauf im Detail

- Die Operationen von LEACH werden in **Runden** unterteilt
- **Advertisement** Phase: Wahl der Cluster-Heads
- **Cluster Set-Up** Phase: Aufbau der Cluster
- **Steady-State** Phase: Normaler Betrieb, Datenübertragung



Advertisement Phase

- Jeder Knoten entscheidet, ob er **Cluster-Head** für die aktuelle Runde r sein wird
 - P = Anteil Cluster-Heads im Netz (muss vorab festgelegt werden)
 - G = Menge der Knoten die in den letzten $1/P$ Runden nicht Cluster-Head waren
 - $T(n)$ = Jeder Knoten n bestimmt Zufallszahl z zwischen 0 und 1 und wird Cluster-Head falls $z < T(n)$

$$T(n) = \begin{cases} \frac{P}{1 - P(r \bmod \frac{1}{P})} & n \in G \\ 0 & n \notin G \end{cases}$$

- → Jeder Knoten wird einmal Cluster-Head innerhalb von $1/P$ Runden
- Cluster-Heads broadcasten ihre Entscheidung in Form einer **Advertisement-Nachricht** an die Nachbarknoten

Cluster-Setup und Steady-State Phase

■ Cluster-Setup

- Knoten entscheiden aufgrund der erhaltenen **Advertisements**, welchem Cluster sie sich zuordnen
- Entscheidung wird dem Cluster-Head mitgeteilt
- Der Cluster-Head sammelt eingehende Meldungen und erstellt einen TDMA-Zeitplan den er allen Knoten im Cluster bekannt gibt

■ Steady-State

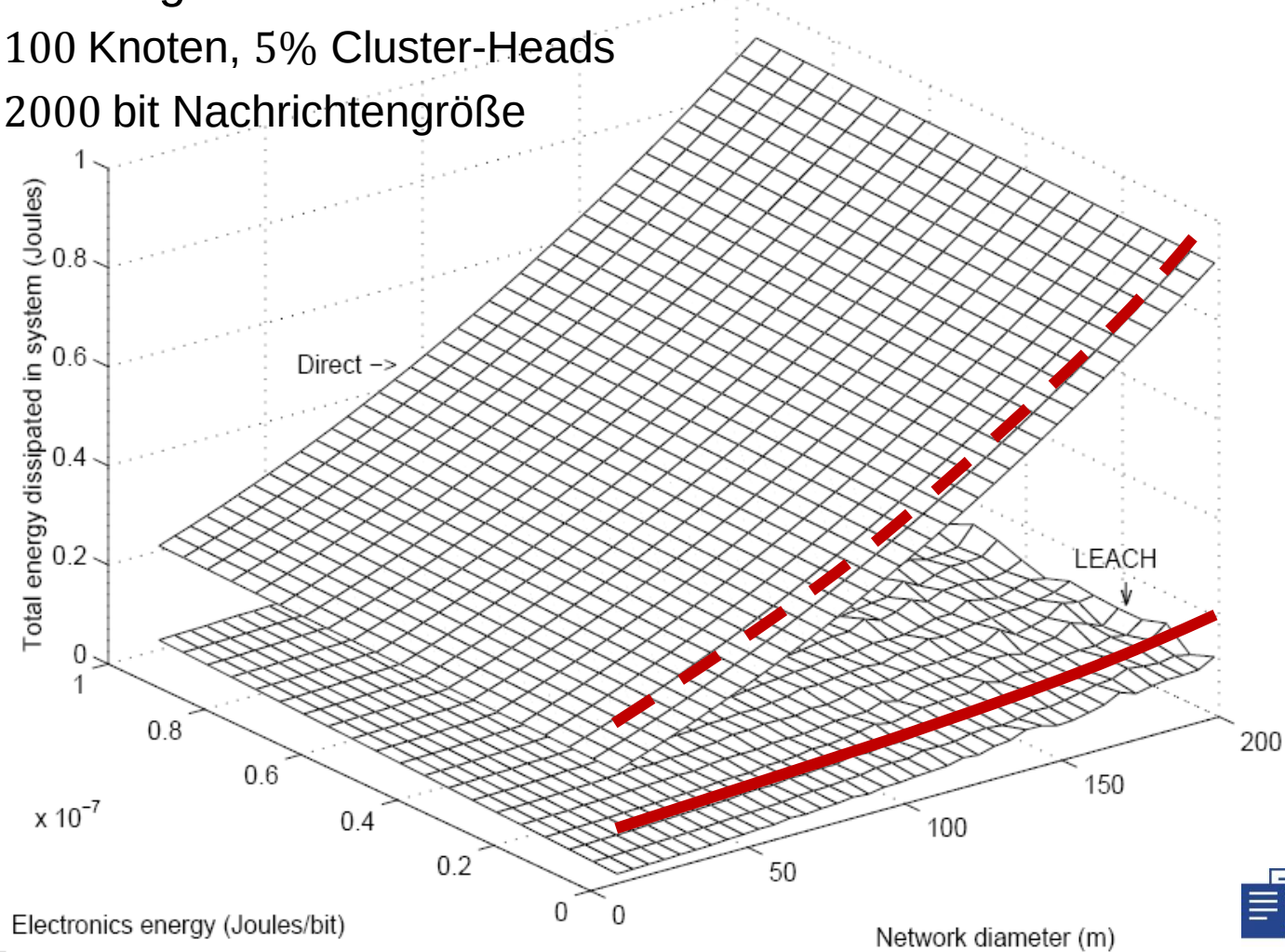
- Knoten im Cluster senden und empfangen gemäß des mitgeteilten Zeitplans und schalten sonst ihren Funkchip aus (→ **Energieeffizienz**)
- Der Cluster-Head kann seinen Funkchip nicht abschalten

- Nach einem festen Zeitintervall d beginnt die nächste Runde $r + 1$

Evaluierung

LEACH verglichen mit direkter Kommunikation aller Knoten mit Basis

- 100 Knoten, 5% Cluster-Heads
- 2000 bit Nachrichtengröße



[HCB00]

LEACH – Fazit

- Verfahren verwendet Clustering zur Topologiekontrolle
- Viele weitere Protokollmechanismen vorhanden
 - TDMA-Zeitplan
→ MAC
 - Zusammenfassen von Messwerten an den Cluster-Heads möglich
→ Aggregation
 - Rotieren der Cluster-Heads zur gleichmäßigen Verteilung des Energiebedarfs
→ Energieeffizienz
- Verbesserungen denkbar bzw. vorhanden
 - Cluster-Head Wahl nicht rein probabilistisch sondern basierend auf Restenergie der Knoten



Die von uns zur Erstellung der Folien genutzte

LITERATUR



- [HCB00] Heinzelman W.R., Chandrakasan A., Balakrishnan H. Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In: Proceedings of the 33rd Hawaii International Conference on System Sciences. S. 174-185. Hawaii, HI, USA. Januar 2000.
- [RaRo00] Ramanathan R., Rosales-Hain R.: Topology Control of Multihop Wireless Networks using Transmit Power Adjustments. In: Proceedings of IEEE Infocom, Tel-Aviv, Israel. S. 404-413. März 2000.
- [SelfOrg] F. Dressler, Vorlesung Selbstorganisation,
<http://www.ccs-labs.org/~dressler/teaching/selbstorganisation-ss10/>
- [Tous80] Toussaint G.: The Relative Neighborhood Graph of a Finite Planar Set. In: Pattern Recognition 12: S. 261-268. 1980.