Wireless Sensor Networks

22nd Lecture

24.01.2007

CoNe
Freiburg

University of Freiburg
Computer Networks and Telematics
Prof. Christian Schindelhauer

Christian Schindelhauer
schindel@informatik.uni-freiburg.de

University of Freiburg
Institute of Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

Options for topology control

- Topology control
  - Control ***node*** activity – deliberately turn on/off nodes
  - Control ***link*** activity – deliberately use/not use certain links

- Topology control
  - ***Flat network*** – all nodes have essentially same role
    - Power control
  - ***Hierarchical network*** – assign different roles to nodes; exploit that to control node/link activity
    - Backbones
    - Clustering

Power control – magic numbers?

- **Question: What is a good power level for a node to ensure "nice" properties of the resulting graph?**
- **Idea: Controlling transmission power corresponds to controlling the number of neighbors for a given node**
- **Is there an "optimal" number of neighbors a node should have?**
  - Is there a "magic number" that is good irrespective of the actual graph/network under consideration?
- **Historically, k=6 or k=8 had been suggested as such "magic numbers"**
  - However, they optimize progress per hop – they do ***not*** guarantee connectivity of the graph!!

  $\rightarrow$ Needs deeper analysis

Controlling transmission range

University of Freiburg
Institute of Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

- **Assume all nodes have identical transmission range r=r(|V|), network covers area A, V nodes, uniformly distr.**
- **Fact: Probability of connectivity goes to zero if:**

$$r(|V|) \leq \sqrt{\frac{(1-\epsilon)A \log |V|}{\pi |V|}}, \text{ for any } \epsilon > 0$$

- **Fact: Probability of connectivity goes to 1 for**

$$r(|V|) \geq \sqrt{\frac{A(\log |V| + \gamma_{|V|})}{\pi |V|}}$$

if and only if $\gamma_{|V|} \to \infty$ with |V|

- **Fact (uniform node distribution, density ρ):**

$$P(\text{G is } k\text{-connected}) \approx \left(1 - \sum_{l=0}^{k-1} \frac{(\rho \pi r^2)^l}{l!} e^{-\rho \pi r^2}\right)$$

Options for topology control

University of Freiburg
Institute of Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

- Topology control
  - Control ***node*** activity – deliberately turn on/off nodes
  - Control ***link*** activity – deliberately use/not use certain links

- Topology control
  - ***Flat network*** – all nodes have essentially same role
    - Power control
  - ***Hierarchical network*** – assign different roles to nodes; exploit that to control node/link activity
    - Backbones
    - Clustering

Hierarchical networks – backbones

- **Idea: Select some nodes from the network/graph to form a *backbone***
  - A connected, minimal, dominating set (MDS or MCDS)
  - Dominating nodes control their neighbors
  - Protocols like routing are confronted with a simple topology – from a simple node, route to the backbone, routing in backbone is simple (few nodes)
- **Dominating Set:**
  - Given an undirected graph G=(V,E)
  - Find a minimal subset $W \subseteq V$ such that for all $u \in W$ there exists $v \in V$ with $\{u,v\} \in V$
- **Problem: MDS is an NP-hard problem**
  - Hard to approximate, and even approximations need quite a few messages
  - Polynomial approximable within c log n for some $c > 0$ only if P=NP
  - Polynomial approximable within a factor of $1 + \log n$.

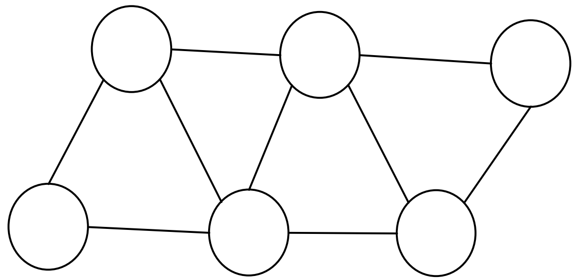
Backbone by growing a tree

University of Freiburg
Institute of Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

- **Construct the backbone as a tree, grown iteratively**

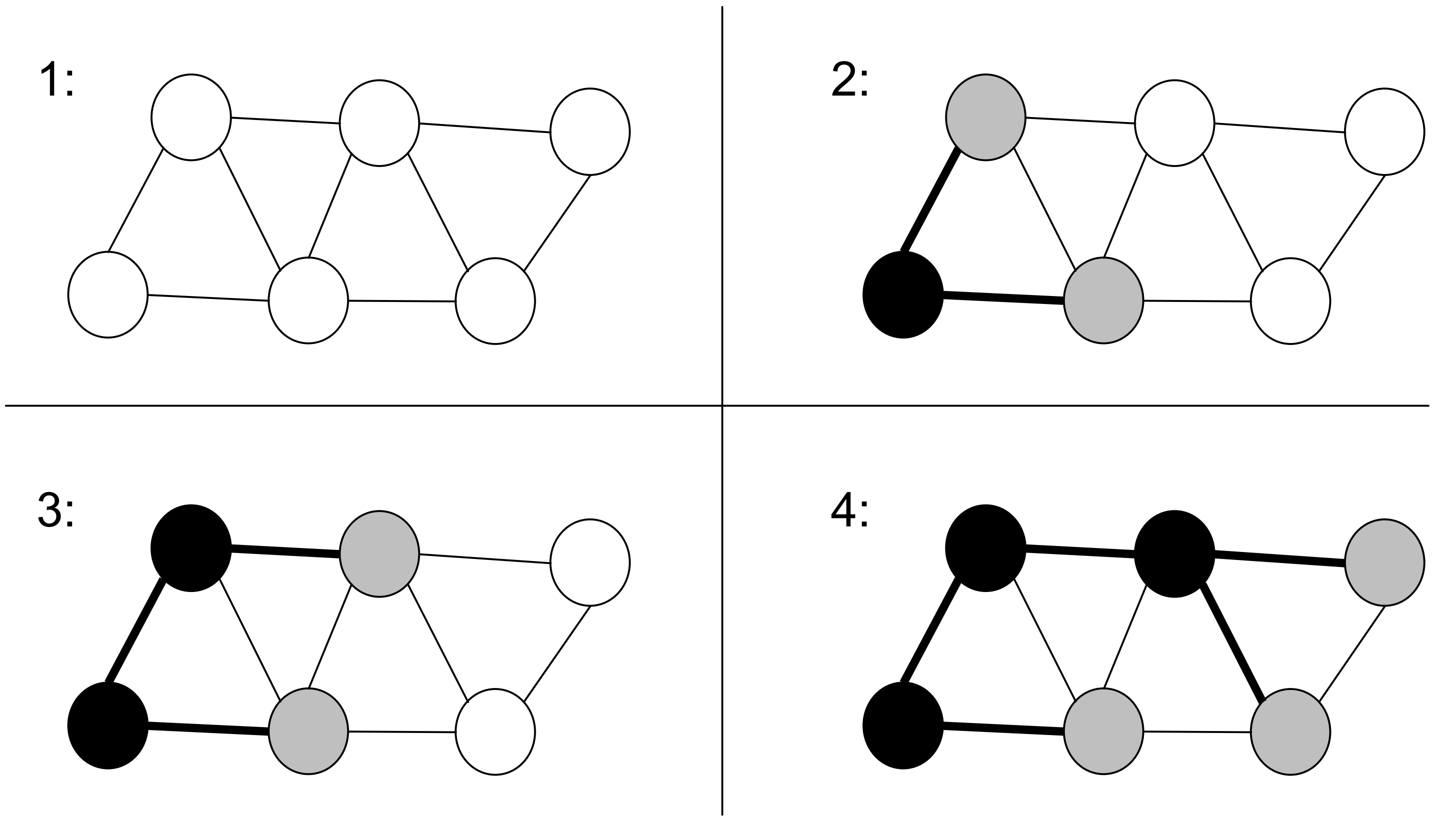
```
initialize all nodes' color to white
pick an arbitrary node and color it grey

while (there are white nodes) {
  pick a grey node v that has white neighbors
  color the grey node v black
  foreach white neighbor u of v {
    color u grey
    add (v,u) to tree T
  }
}
```



Backbone by growing a tree – Example

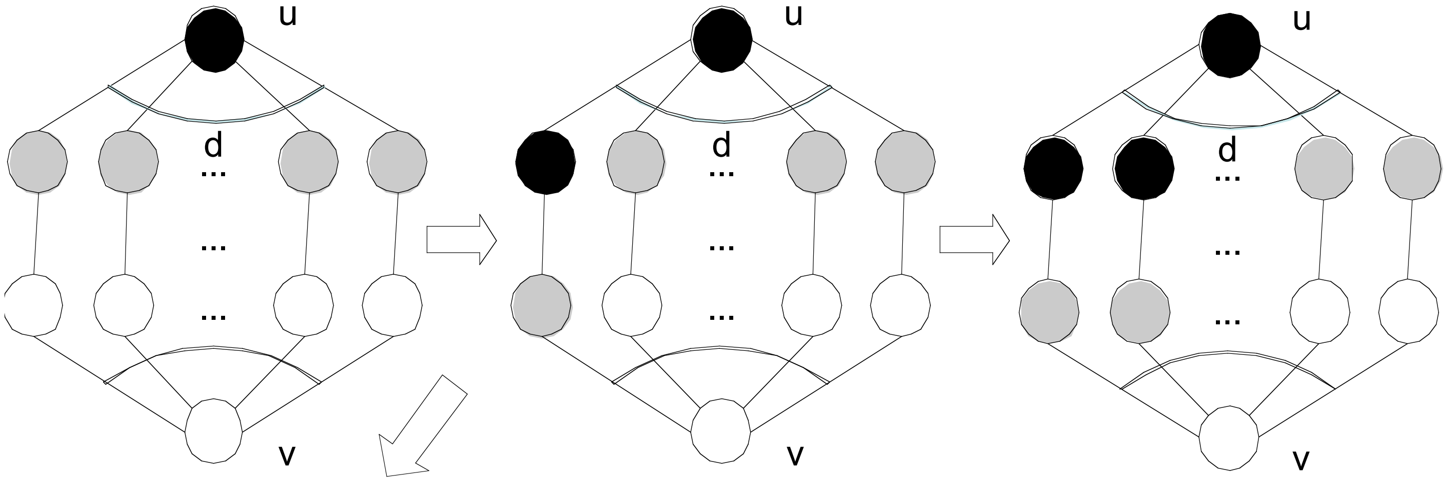
University of Freiburg
Institute of Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer



Problem: Which gray node to pick?

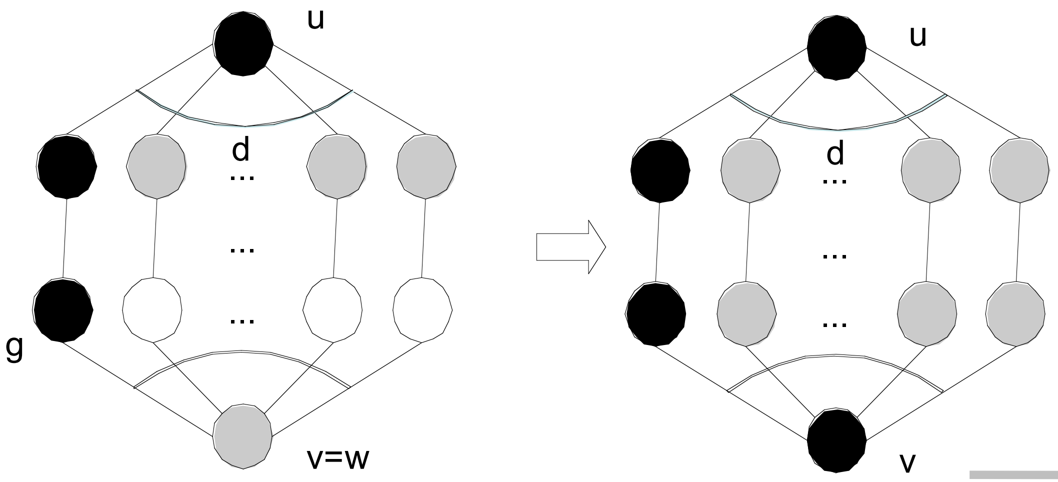
University of Freiburg
Institute of Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

- **When blindly picking any gray node to turn black**
  - resulting tree can be very bad



Solution:
Look ahead!
Here,
one step suffices

Look-ahead using nodes g and w



Performance of tree growing with look ahead

University of Freiburg
Institute of Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

- **Dominating set obtained by growing a tree with the look ahead heuristic is at most a factor 2(1+ H(Δ)) larger than MDS**
  - H(·) harmonic function, $H(k) = \sum_{i=1}^{k} 1/i \leq \ln k + 1$
  - Δ is maximum degree of the graph

- **It is automatically connected**

- **Can be implemented in a distributed fashion as well**

Start big, make lean

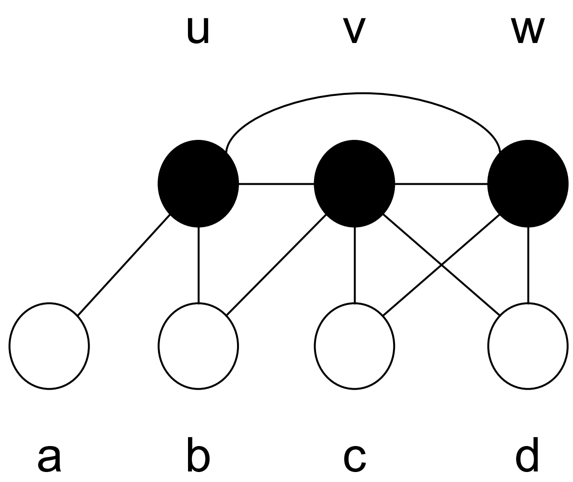
- **Idea: start with some, possibly large, connected dominating set, reduce it by removing unnecessary nodes**
- **Initial construction for dominating set**
  - All nodes are initially white
  - Mark any node black that has two neighbors that are not neighbors of each other (they might need to be dominated)

  $\rightarrow$ Black nodes form a connected dominating set (proof by contradiction); shortest path between ANY two nodes only contains black nodes

- **Needed: Pruning heuristics**

Pruning heuristics

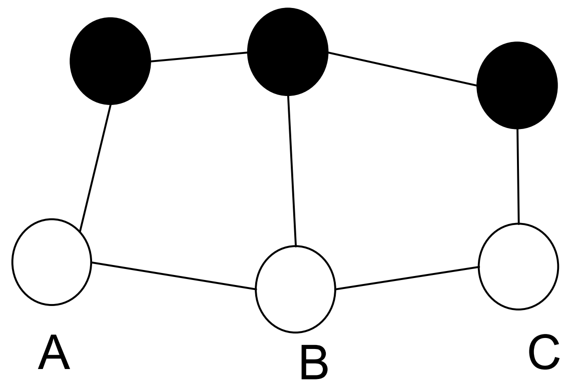
- **Heuristic 1: Unmark node v if**
  - Node v and its neighborhood are included in the neighborhood of some node marked node u (then u will do the domination for v as well)
  - Node v has a smaller unique identifier than u (to break ties)
- **Heuristic 2: Unmark node v if**
  - Node v's neighborhood is included in the neighborhood of two marked neighbors u and w
  - Node v has the smallest identifier of the tree nodes
- **Nice and easy, but only linear approximation factor**



University of Freiburg
Institute of Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

One more distributed backbone heuristic: Span

- **Construct backbone, but take into account need to carry traffic – preserve capacity**
  - Means: If two paths could operate without interference in the original graph, they should be present in the reduced graph as well
  - Idea: If the stretch factor (induced by the backbone) becomes too large, more nodes are needed in the backbone
- **Rule: Each node observes traffic around itself**
  - If node detects two neighbors that need three hops to communicate with each other, node joins the backbone, shortening the path
  - Contention among potential new backbone nodes handled using random backoff



Thank you

and thanks to Holger Karl and Rajmohan Rajamaran for some slides

CoNe Freiburg

University of Freiburg
Computer Networks and Telematics
Prof. Christian Schindelhauer

Wireless Sensor Networks
Christian Schindelhauer
schindel@informatik.uni-freiburg.de

22nd Lecture
24.01.2007