Mobile Ad Hoc Networks

Theory of Data Flow and Random Placement

3rd Week

02.05.-04.05.2007

CoNe Freiburg

University of Freiburg
Computer Networks and Telematics
Prof. Christian Schindelhauer

Christian Schindelhauer
schindel@informatik.uni-freiburg.de

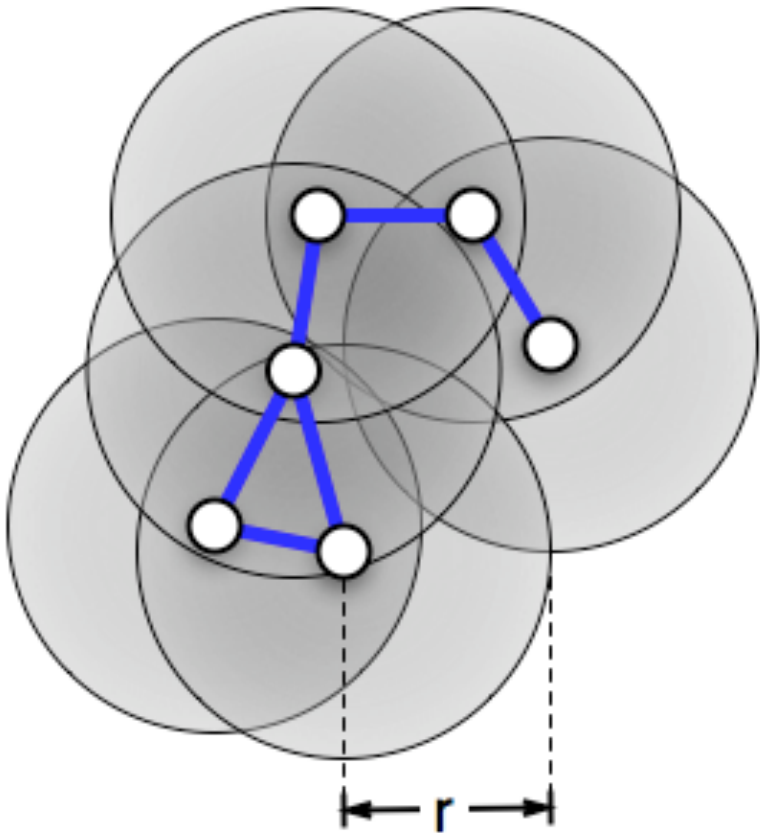
Unit Disk Graphs

- **Motivation:**
  - Received Signal Strength decreases proportionally to $d^{-\gamma}$,
    - where γ is the path loss exponent
  - Connections only exists if the signal/noise ratio is beyond a threshold
- **Definition**
  - Given a finite point set V in $\mathbf{R}^2$ or $\mathbf{R}^3$,
  - then a Unit Disk Graph with radius r G=(V,E) of the point set is defined by the undirected edge set:

$$E = \{\{u,v\} \mid ||u,v||_2 \leq r\}$$

  - where $||u,v||_2$ is the Euclidean distance:

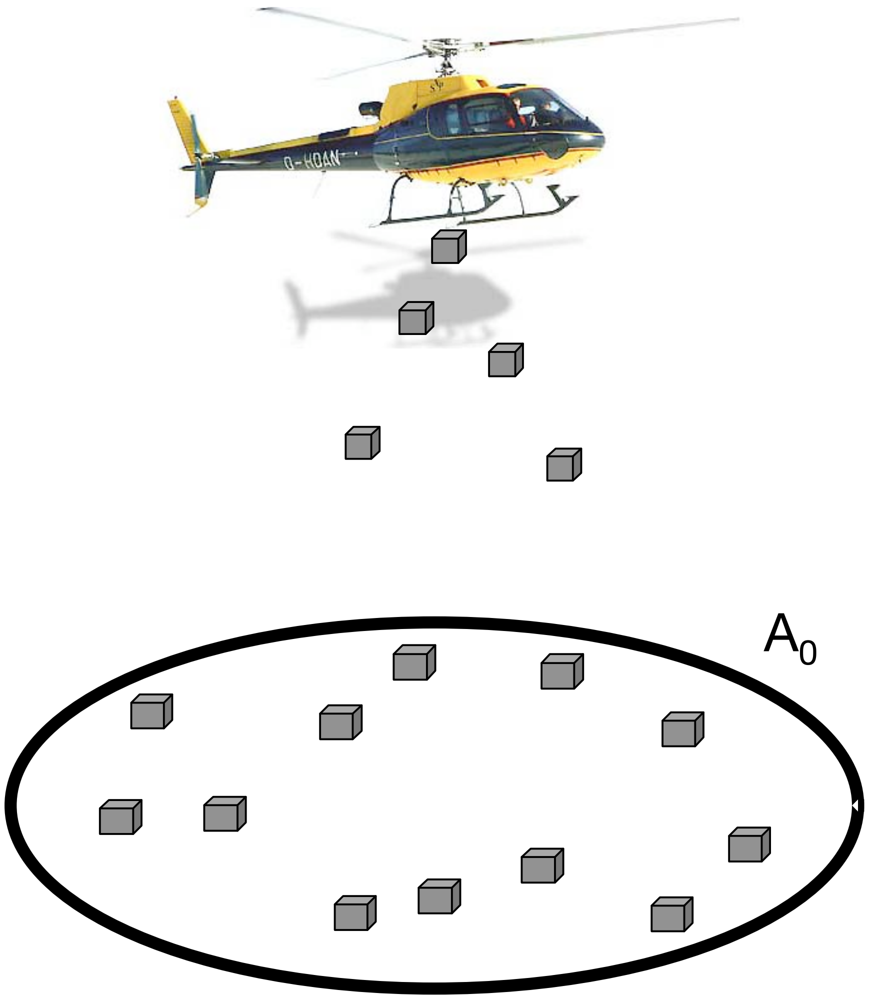
$$||u,v||_2 = \sqrt{(u_x - v_x)^2 + (u_y - v_y)^2}$$



Random Placement Model

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

- **Motivation**
  - Throwing nodes from a plane
  - Natural processes lead to a random placement
- **Definition**
  - A set of points is placed randomly in an area A_0 if every position occurs with equal probability, i.e.
  - the probability density function (pdf) f(x) is a constant



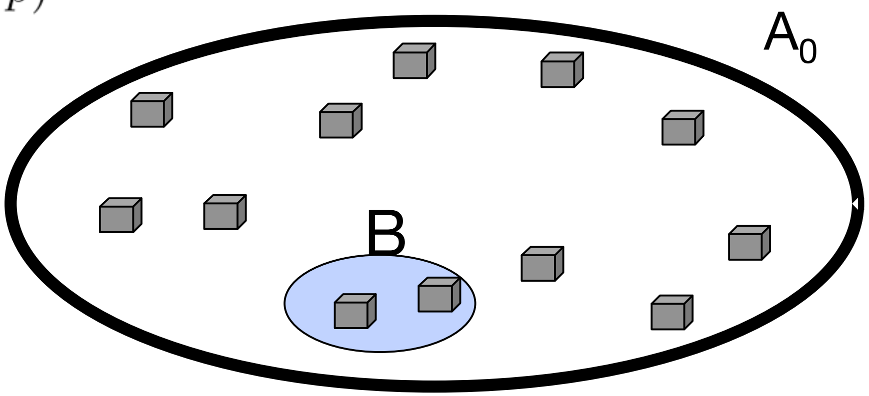
Properties of Random Placement

- **The probability that a node falls in a specific area B of the overall area A_0 is**

$$Pr[\text{a node falls in } B] = \frac{|B|}{|A_0|}$$

  - where |B| denotes the area of B
- **Lemma**
  - The probability that k of n nodes fall in an area B with p = |B|/$|A_0|$ is

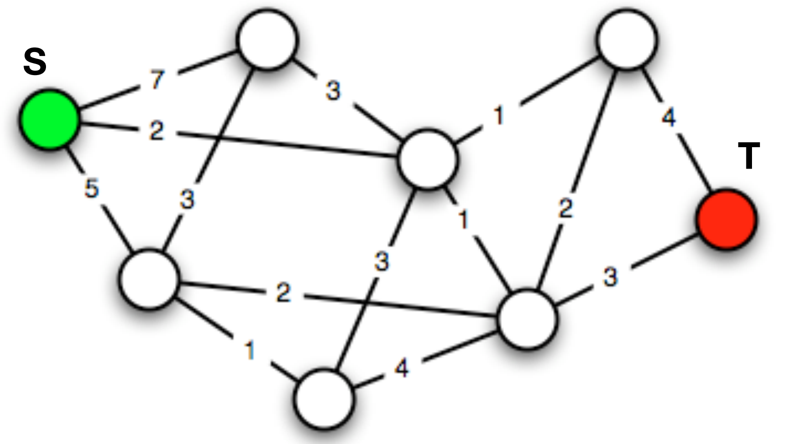
$$Pr\left[\begin{array}{c} k \text{ of } n \text{ nodes} \\ \text{fall in area } B \end{array}\right] = \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k}$$

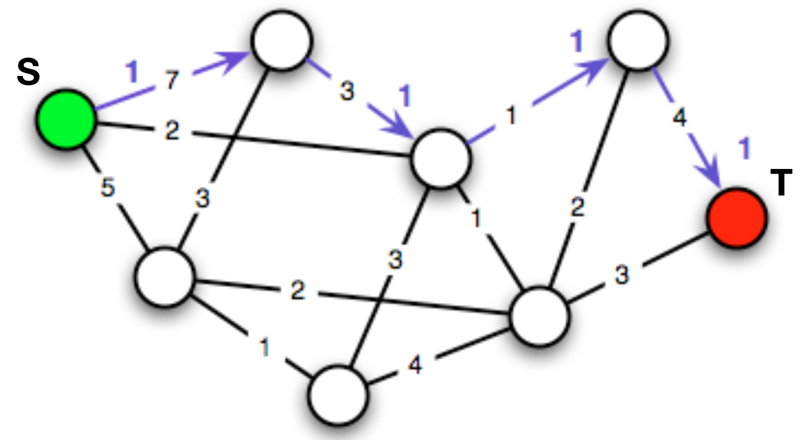


Data Flow in Networks

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

- **Motivation:**
  - Optimize data flow from source to target
  - Avoid bottlenecks
- **Definition:**
  - **(Single-commodity) Max flow problem**
  - Given
    - a graph G=(V,E)
    - a capacity function w: E → $\mathbf{R}^+_0$,
    - source set S and target set T
  - Find a maximum flow from S to T
- **A flow is a function f : E →** R^+_0 **with**
  - for all e ∈ E: f(e) ≤ w(e)
  - for all e ∉ E: f(e) = 0
  - for all u,v ∈ V: f(u,v)≥0
  - $\forall u \in V \setminus (S \cup T)$
$$\sum_{v \in V} f(v,u) = \sum_{v \in V} f(u,v)$$
- **The size of a flow is:** $$\sum_{u \in S} \sum_{v \in V} f(u,v)$$

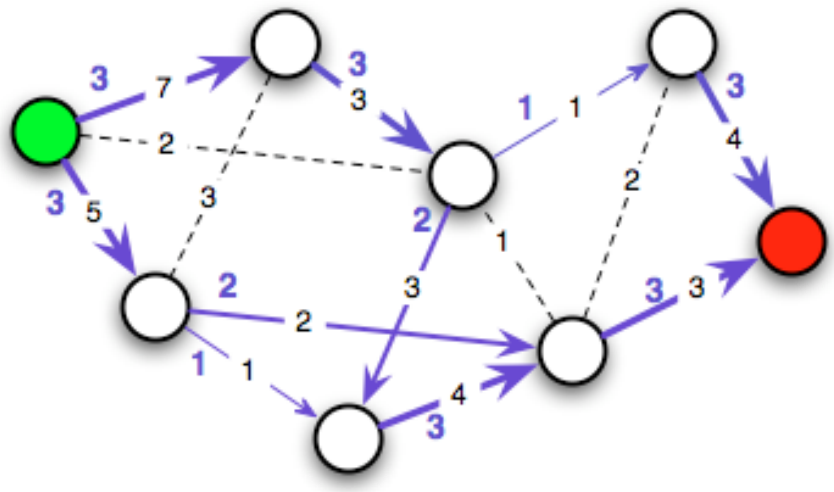




Finding the Max Flow

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

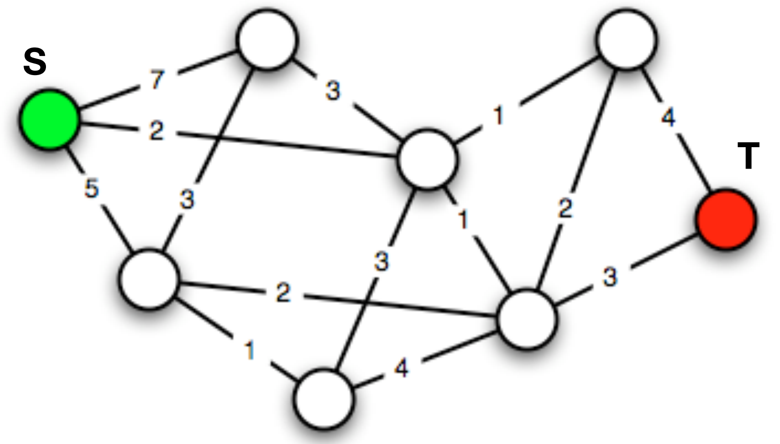
- **In every natural pipe system the maximum flow is computed by nature**
- **Computer Algorithms for finding the max flow:**
  - Linear Programming
    - The flow equalities are the constraints of a linear optimization problem
    - Use Simplex (or ellipsoid method) for solving this linear equation system
  - Ford-Fulkerson
    - As long there is an open path (a path which improves the flow) increase the flow on this path
  - Edmonds-Karp
    - Special case Ford-Fulkerson
    - Use Breadth-First-Search to find the paths

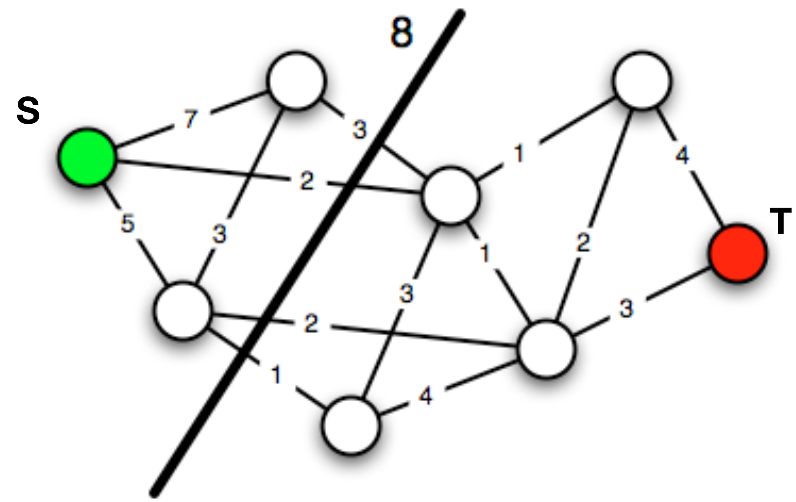


Min Cut in Networks

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

- **Motivation:**
  - Find the bottleneck in a network
- **Definition:**
  - **Min cut problem**
  - Given
    - a graph G=(V,E)
    - a capacity function w: E → $\mathbf{R}^{+}_{0}$,
    - source set S and target set T
  - Find a minimum cut between S and T
- **A cut C is a set of edges such that**
  - there is no path from any node in S to any node in T
- **The size of a cut C is:** $\sum_{e \in C} w(e)$

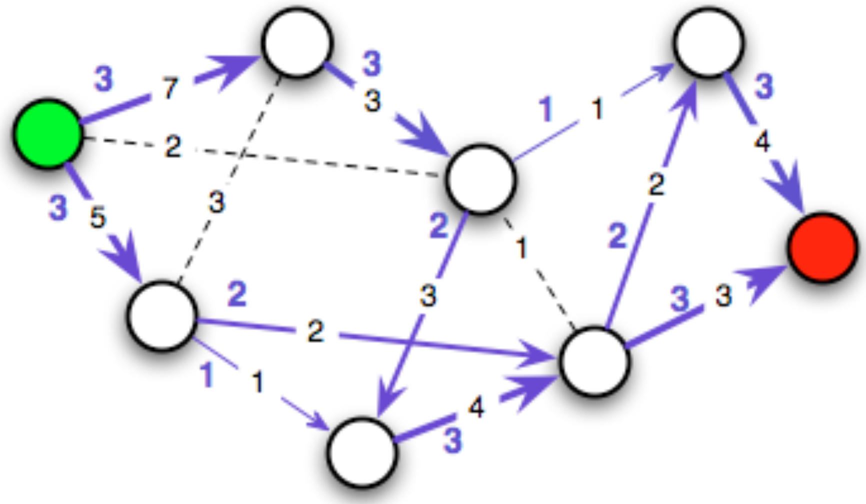




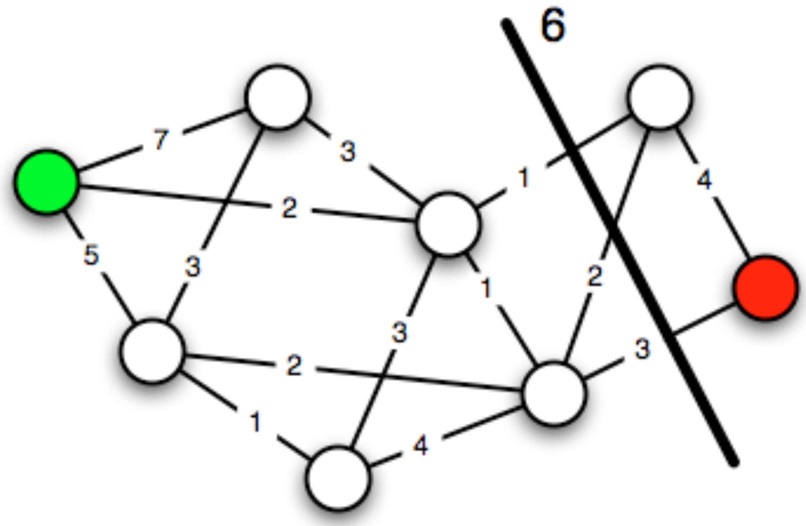
Min-Cut-Max-Flow Theorem

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

- **Theorem**
  - For all graphs, all capacity functions, all sets of sources and sets of targets
  - **the minimum cut equals the maximum flow.**

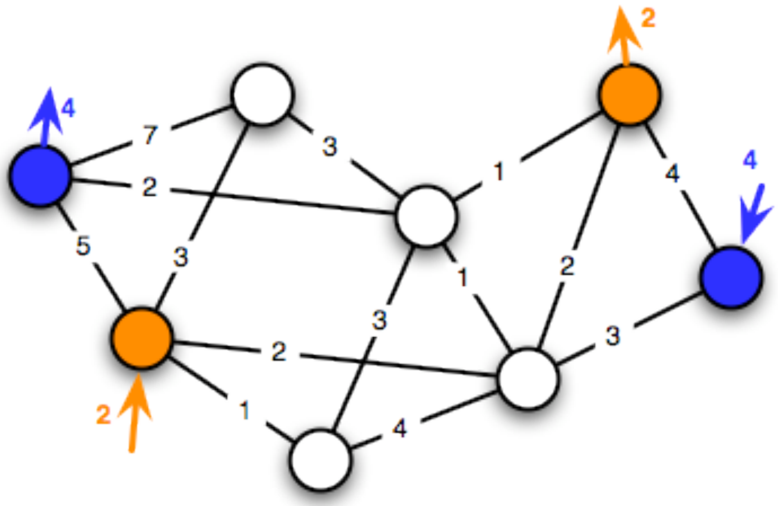


- **Algorithms for minimum cut**
  - –like algorithms for max flow.



Multi-Commodity Flow Problem

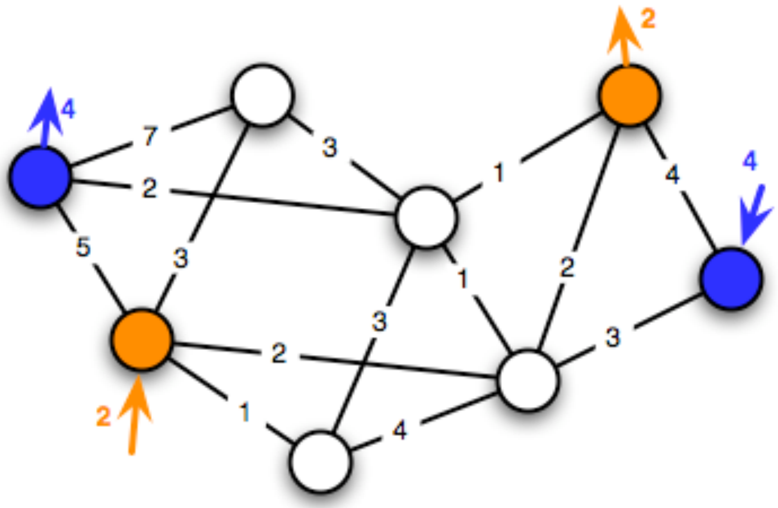
- **Motivation:**
  - Theoretical model of all communication optimization for point-to-point communication with capacities
- **Definition**
  - **Multi-commodity flow problem**
  - Given
    - a graph G=(V,E)
    - a capacity function w: E → $\mathbf{R}^{+}_{0}$,
    - commodities K_1, .., K_k:
      - $K_i=(s_i,t_i,d_i)$ with
      - s_i is the source node
      - t_i is the target node
      - d_i is the demand

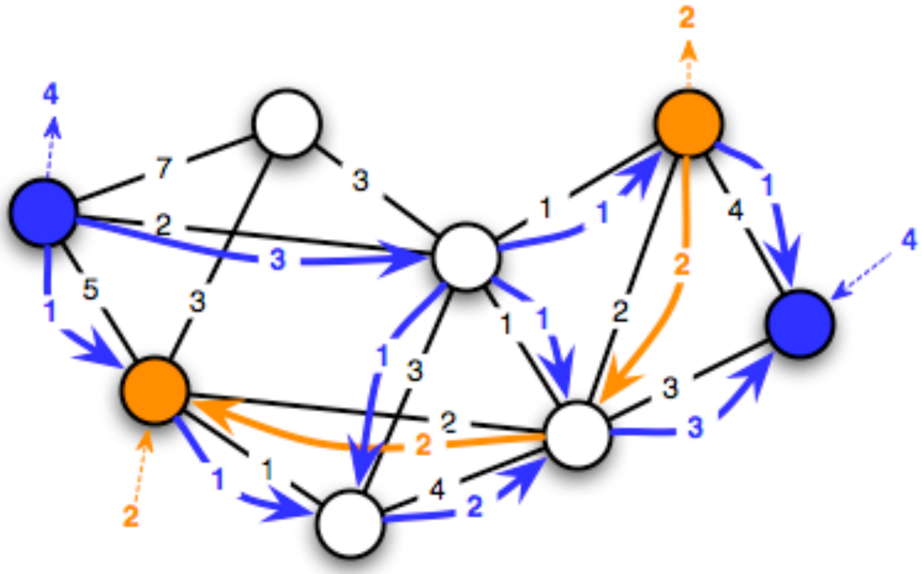


- **Find flows $f_1,f_2,\ldots,f_k$ for all commodities obeying**
  - Capacity: $\sum_{i=1}^{k} f_i(u,v) \le w(u,v)$
  - Flow property: $\forall v \notin \{s_i,t_i\} \;:\; \sum_{u\in V} f_i(u,v) = \sum_{u\in V} f_i(v,u)$
  - Demand: $\sum_{v\in V} f_i(s_i,v) = \sum_{u\in V} f_i(u,t_i) = d_i$

Solving Multi-Commodity Flow Problems

- **The Multi-Commodity Flow Problem can be solved by linear programming**
  - Use equality as constraints
  - Use Simplex or Ellipsoid Algorithm
- **There exist weakened versions of min-cut-max-flow theorems**





Minimum Density for Connectivity

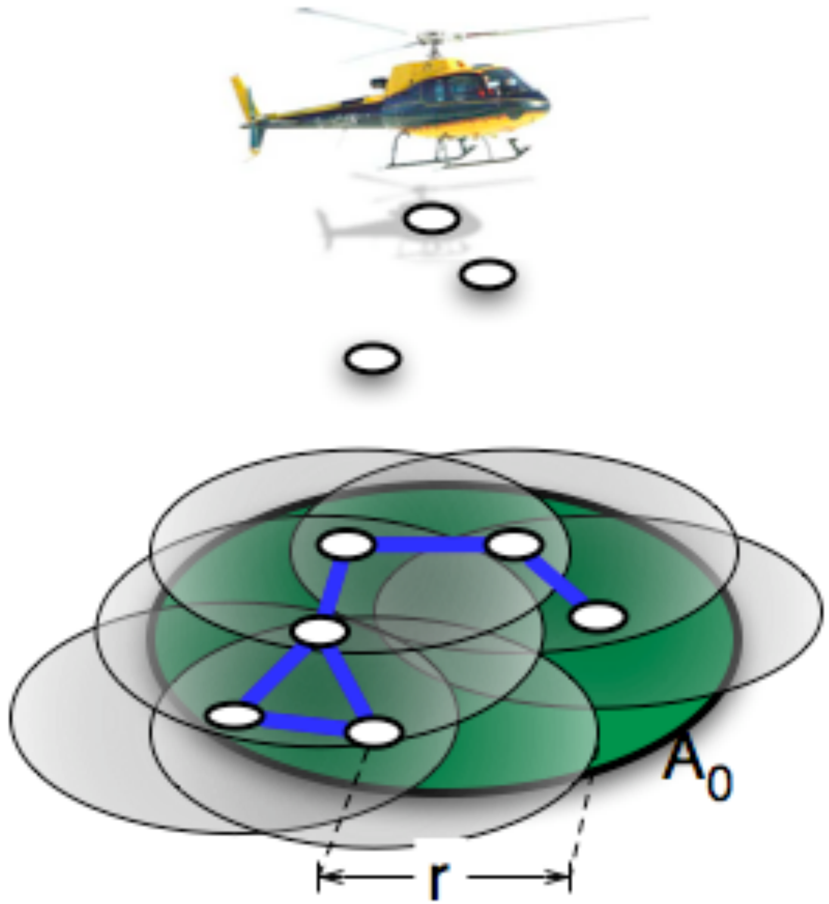
University of Freiburg
Institute of Computer Science
Computer Networks and Telematics
Prof. Christian Schindelhauer

- **Gupta, Kumar**
  - Critical Power for Asymptotic Connectivity in Wireless Networks
- **Motivation:**
  - How many nodes need to be placed to achieve a connected UDG (unit-disk graph)
- **Theorem**
  - In the square area A_0 it is necessary and sufficient to uniformly random place n nodes to achieve a connected UDG where

$$c \cdot \pi r^2 \cdot n = |A_0| \log n$$

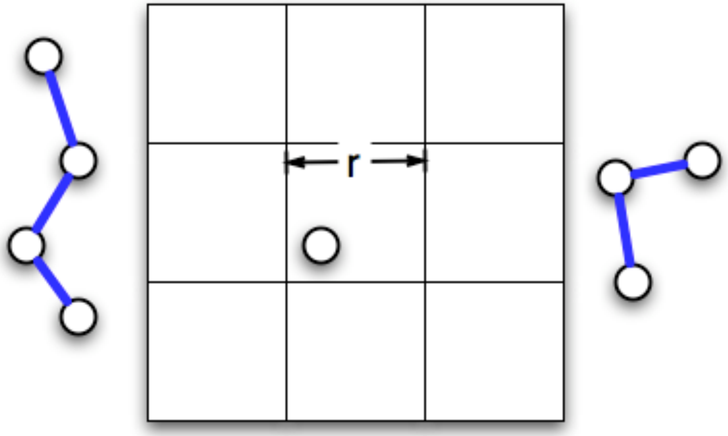
  - for some constant factor c.
- **Equivalent description:**

$$\Theta\left(\frac{n}{\log n}\right) = \frac{|A_0|}{r^2}$$



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

Why so Many Nodes?



- **Sufficient condition for unconnectedness**
  - At least one node in a square of edge length r
  - 8 neighbored squares are empty
- **Probability for none of the n nodes in surrounding squares:**

$$\left(1 - \frac{8r^2}{|A_0|}\right)^n$$

- **Note that for x ∈ [0,0.75]:**

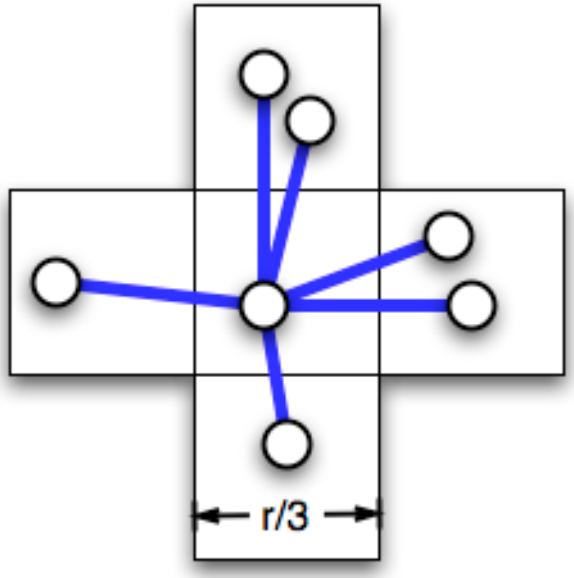
$$e^{-2x} \leq (1-x) \leq e^{-x}$$

- **Therefore (for large enough A_0)**

$$\left(1 - \frac{8r^2}{|A_0|}\right)^n \geq e^{-\frac{16r^2 n}{|A_0|}}$$

- **The expected number of such isolated nodes is at least** $n \cdot e^{-\frac{16r^2 n}{|A_0|}}$
- **If** $r^2 = \omega\left(\frac{|A_0| \ln n}{n}\right)$ **then the expected number of unconnected nodes is at least 1**

Are so Many Nodes Sufficient?



- **Sufficient property of connectivity**
  - In the adjacent squares of edge length r/3 is at least one node
- **Probability that at least one node is in such a square:**

$$1 - \left(1 - \frac{r^2}{9|A_0|}\right)^n$$

- **Choose**

$$r^2 = c \cdot \frac{|A_0| \ln n}{n}$$

- **Then the above probability is:**

$$1 - \left(1 - \frac{c \ln n}{9n}\right)^n \geq 1 - e^{-\frac{c}{9}\ln n} = 1 - n^{-\frac{c}{9}}$$

- **Choose c>9**
  - then the chance of such an occupied neighbored square is bounded by $o(n^{-1})$
  - Multiplying this probability with 4n for all neighbored squares gives an upper bound on the probability that each node does not have neighbors to the four sides
- **Then, the error probability is bounded by o(1)**

Network Flow in Random Unit Disk Graphs

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

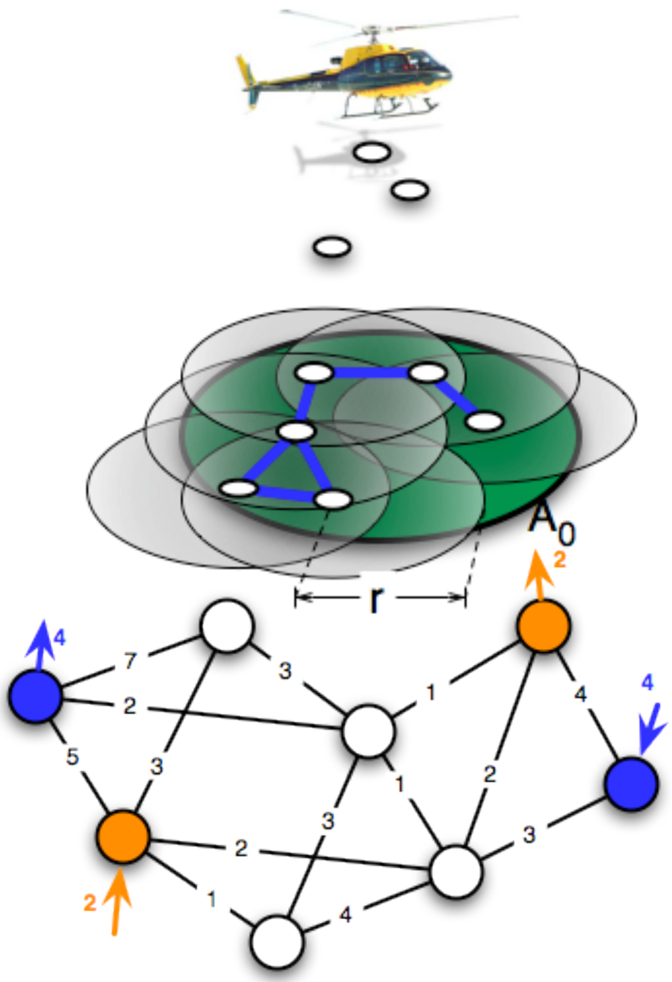
- **Motivation:**
  - What is the communication capability of the network
- **Theorem**
  - Assume that in the square area A_0 if n nodes are uniformly random placed where

$$\Theta\left(\frac{n}{\log n}\right) = \frac{|A_0|}{r^2}$$

  - Assume that there is a multi-commodity flow problem in UDG where each node sends to each other node a packet of size 1
  - Then each demand d can be satisfied if the capacity of each edge is

$$O\left(\frac{W}{\sqrt{n \log n}}\right)$$

  - where $W=n^2$ is the sum of all packets

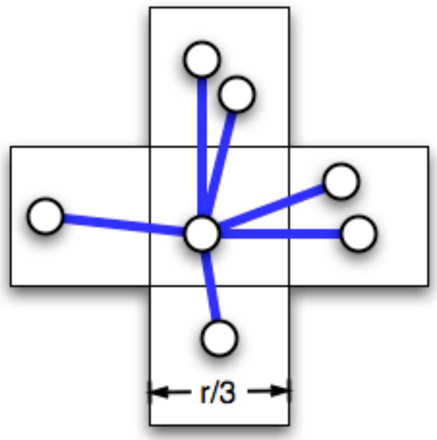


Proof Sketch

- **First observation:**
  - for $\Theta\left(\frac{n}{\log n}\right) = \frac{|A_0|}{r^2}$
  - the random placement leads to a grid like structure where each cell of cell length r/3
- **Second observation:**
  - The network is mainly a grid with m x m cells, where

$$m = \Theta\left(\sqrt{\frac{n}{\log n}}\right)$$

  - On the average each cell has log n nodes and has this number edges to the neighbored cells
  - In a grid such a demand can be routed with capacity n^2/m (horizontal or vertical cut is bottleneck)
  - In this network the minimum cut is now m log n = $(n \log n)^{1/2}$
  - The multicommodity flow is therefore $W/(n \log n)^{1/2}$



Discussion

- **Randomly placed connected UDGs need an overhead of a factor of O(log n) nodes**
  - to become connected
- **Then the networks behave like grids**
  - up to some polylogarithmic factor
- **The bottleneck of grids is the width**
  - in the optimal case of square-like formations this is $n^{1/2}$.
- **If the overhead of a factor O(log n) is not achieved**
  - then the randomly placed UDG is not connected
- **This is another case of the coupon-collector problem**
  - How many cards do you need to collect until you possess each of n coupons

Thank you!

CoNe
Freiburg

University of Freiburg
Computer Networks and Telematics
Prof. Christian Schindelhauer

Mobile Ad Hoc Networks
Christian Schindelhauer
schindel@informatik.uni-freiburg.de

3rd Week
02.05.2007