Fault-Tolerant Power Assignment and Backbone in Wireless Networks

by

Paz Carmi, Matthew J. Katz, Michael Segal and Hanan Shpungin

Outline

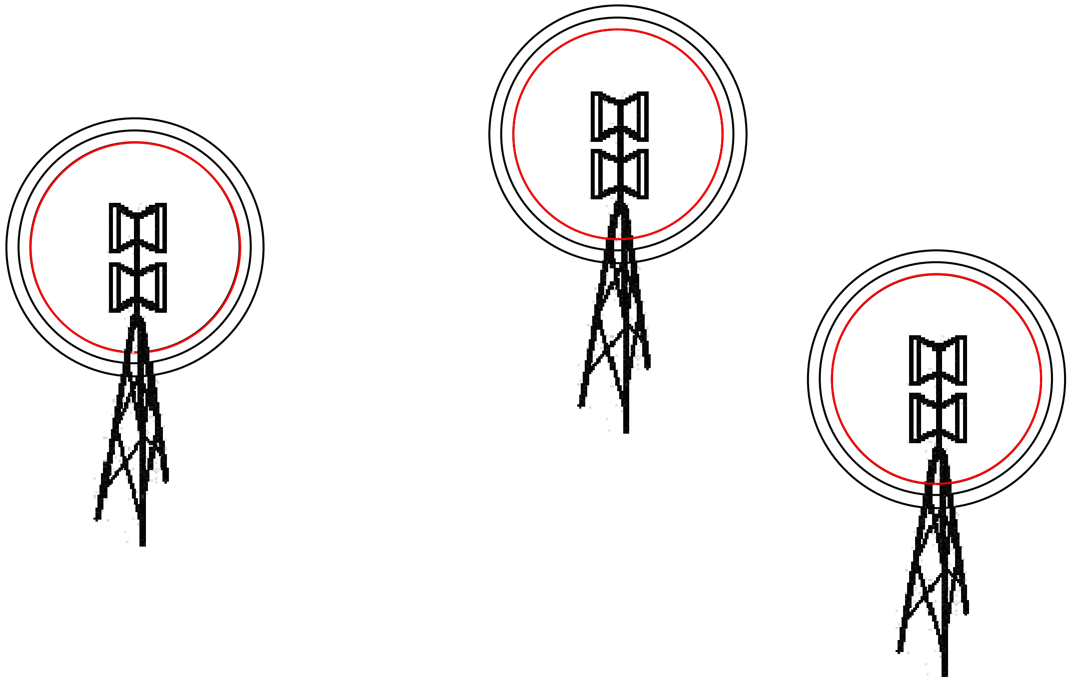
- Introduction
- Model & Problems
- Fault-Tolerant Power Assignment
- Connected Backbone Power Assignment
- Summary

Outline

- Introduction
- Model & Problems
- Fault-Tolerant Power Assignment
- Connected Backbone Power Assignment
- Summary

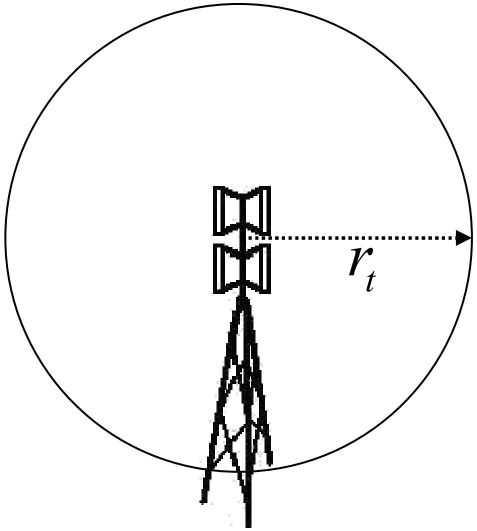
Introduction

- **Wireless Ad-Hoc Network**
  - Set of transceivers communicating by radio



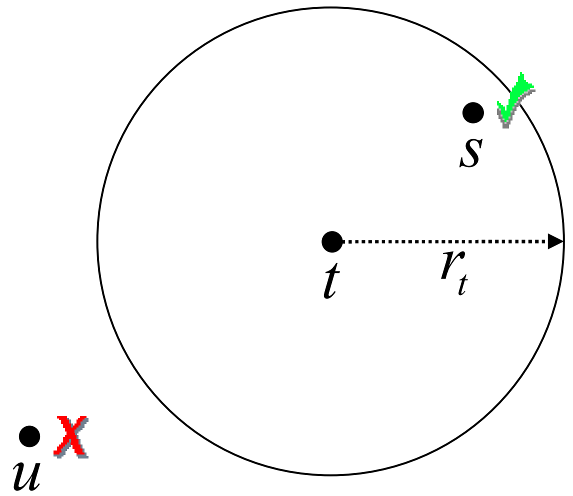
Introduction

- **Wireless Ad-Hoc Network**
  - Each transceiver has a transmission power $p(t)$ which results in a transmission range r_t



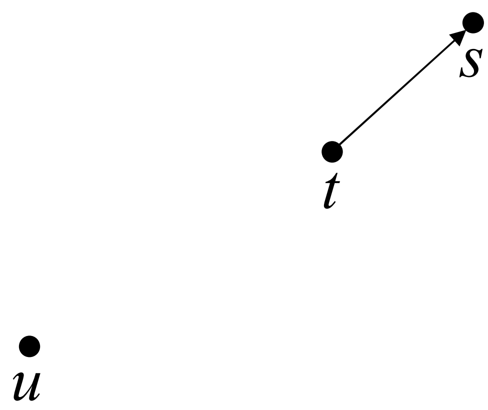
Introduction

- **Wireless Ad-Hoc Network**
  - Transceiver s receives transmission from t only if $d(t, s) \le r_t$



Introduction

- **Wireless Ad-Hoc Network**
  - As a result a directed communication graph is induced

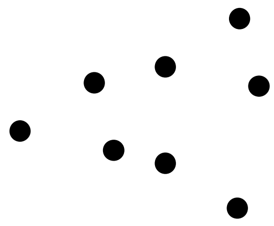


Outline

- Introduction
- Model & Problems
- Fault Tolerant Power Assignment
- Connected Backbone Power Assignment
- Summary

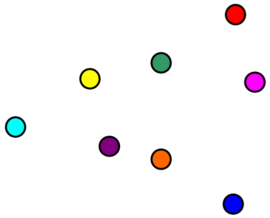
Model & Problems

- **Definition**
  - A set T of n transceivers $t_1, t_2, \ldots, t_n$



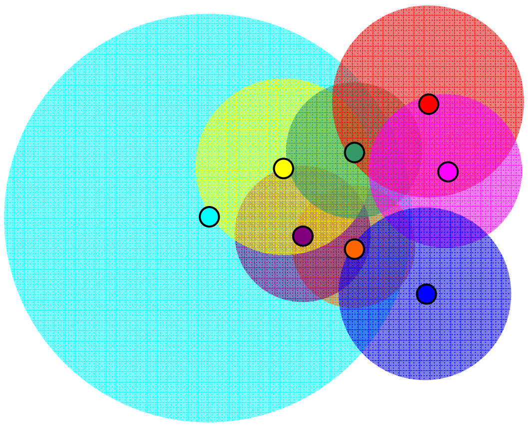
Model & Problems

- **Definition**
  - A set T of n transceivers $t_1, t_2, \ldots, t_n$
  - $A = A(T) = \{p(t) \mid t \in T\}$ is the power assignment



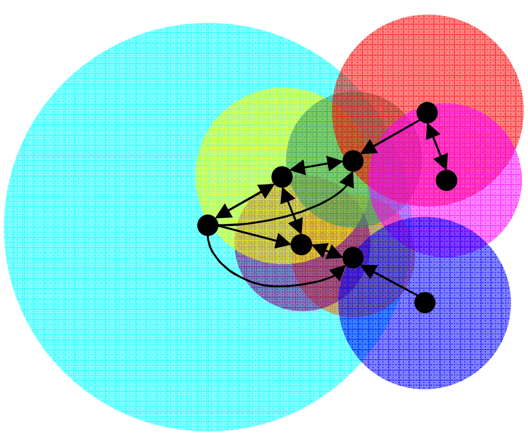
Model & Problems

- **Definition**
  - A set T of n transceivers $t_1, t_2, \ldots, t_n$
  - $A = A(T) = \{p(t) \mid t \in T\}$ is the power assignment



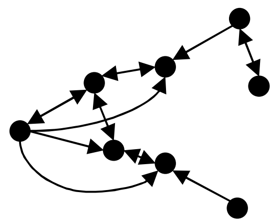
Model & Problems

- **Definitions**
  - A set T of n transceivers $t_1, t_2, \ldots, t_n$
  - $A = A(T) = \{p(t) \mid t \in T\}$ is the power assignment
  - $H_A = (T, E_A)$ is the communication graph



Model & Problems

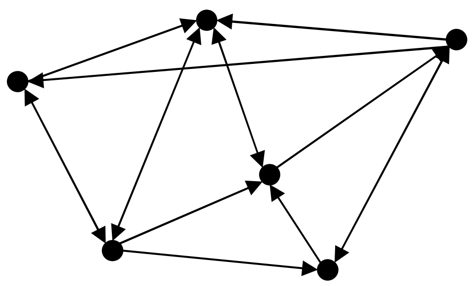
- **Definitions**
  - A set T of n transceivers $t_1, t_2, \ldots, t_n$
  - $A = A(T) = \{p(t) \mid t \in T\}$ is the power assignment
  - $H_A = (T, E_A)$ is the communication graph
  - $C_A = \sum_{t \in T} p(t) = \sum_{t \in T} r_t^{\alpha}$ is the cost of the assignment



Model & Problems

- **Definitions**
  - A graph $G=(V,E)$ is *k-vertex-connected* if for any two nodes $u,v \in V$ there exist k-vertex-disjoint paths connecting u to v

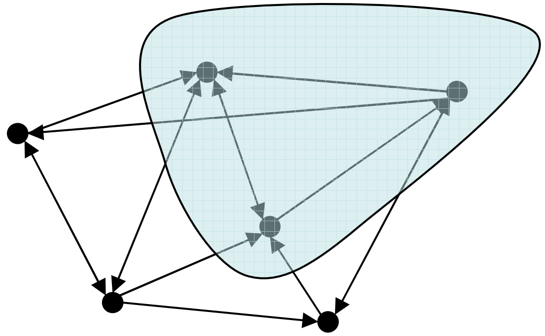
2-vertex-connected



Model & Problems

- **Definitions**
  - For graph $G=(V,E)$, a subset $D \subseteq V$ is a *connected backbone* if G restricted to D is strongly connected and for each $t \in T \setminus D$ there exists $u \in D$ so that $e=(u,t) \in E$

Connected backbone



Model & Problems

- **Problem 1** (*k*-vertex-connectivity)

  Input: A set T of transceivers, and a parameter $k > 1$

  Output: A power assignment $A(T)$ with minimal possible cost C_A, where H_A is *k-vertex connected*

Model & Problems

- **Problem 1** (k-vertex-connectivity)

Input: A set T of transceivers, and a parameter $k > 1$

Output: A power assignment $A(T)$ with minimal possible cost C_A, where H_A is *k-vertex connected*

$O(k)$**-approximation algorithm**

Model & Problems

- **Problem 2** (connected backbone)

Input: A set T of transceivers

Output: A subset D of T and a power assignment $A(D)$ with minimal possible cost C_A, where H_A (restricted to D) is *strongly connected*, and for each $t \in T \setminus D$, there exists $u \in D$, such that $d(u,t) \leq r_u$

Model & Problems

- **Problem 2** (connected backbone)

  Input: A set T of transceivers

  Output: A subset D of T and a power assignment $A(D)$ with minimal possible cost C_A, where H_A (restricted to D) is *strongly connected*, and for each $t \in T \setminus D$, there exists $u \in D$, such that $d(u,t) \le r_u$

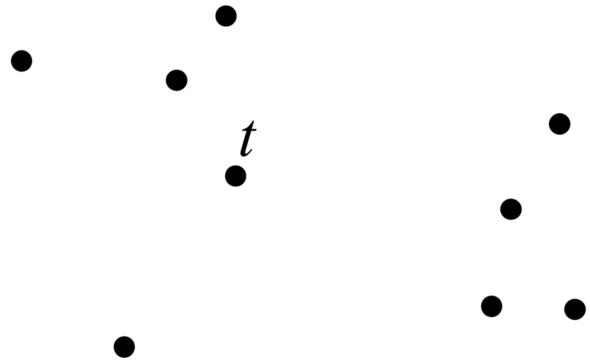
Constant-factor approximation algorithm in $O(n \log n)$

Outline

- Introduction
- Model & Problems
- Fault-Tolerant Power Assignment
- Connected Backbone Power Assignment
- Summary

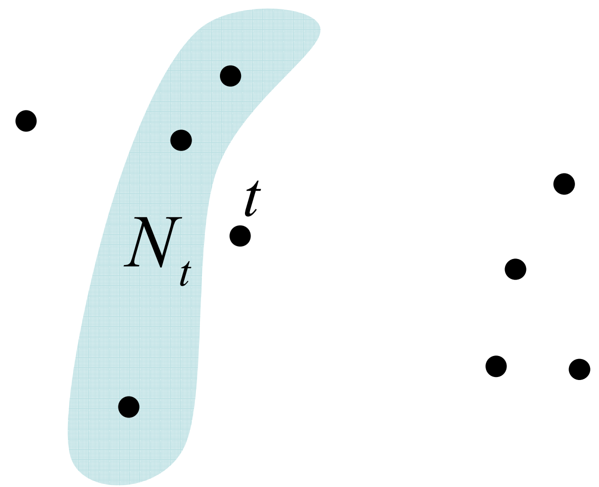
Fault-Tolerant Power Assignment

- **Definitions**
  - For each $t \in T$, let $N_t \subseteq T$ be a set of k closest nodes to t



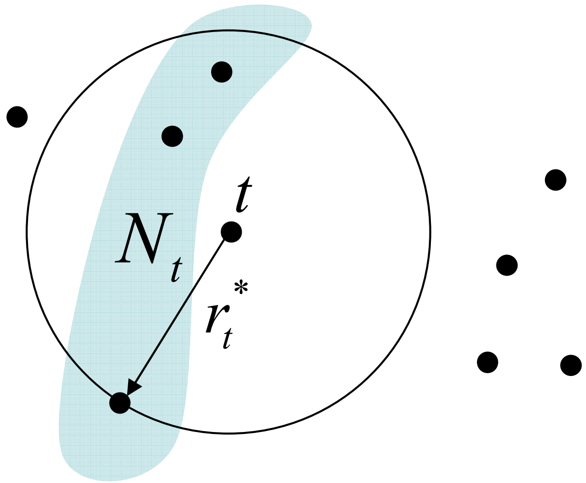
Fault-Tolerant Power Assignment

- **Definitions**
  - For each $t \in T$, let $N_t \subseteq T$ be a set of k closest nodes to t



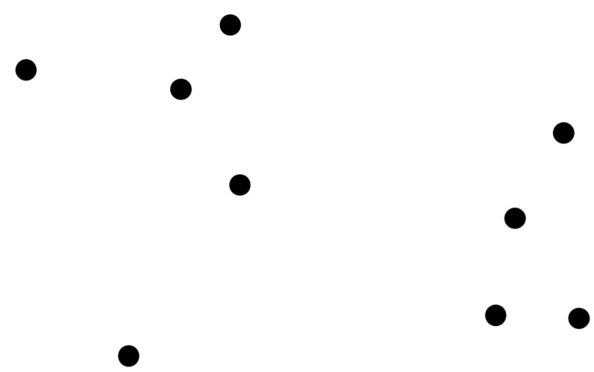
Fault-Tolerant Power Assignment

- **Definitions**
  - For each $t \in T$, let $N_t \subseteq T$ be a set of k closest nodes to t
  - Let $r_t^* = \max_{t' \in N_t} d(t, t')$



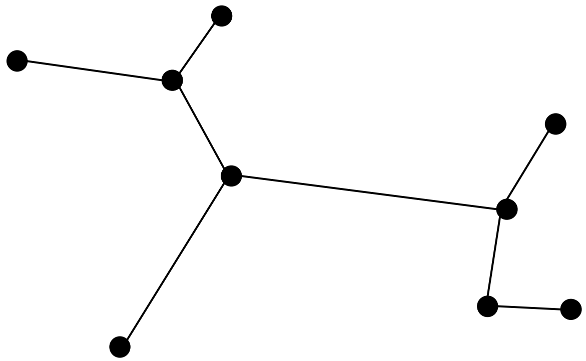
Fault-Tolerant Power Assignment

- **The algorithm**
  - Assign each $t \in T$ the range r_t^* (denote $A_k^{'}$)
  - Compute an MST of T



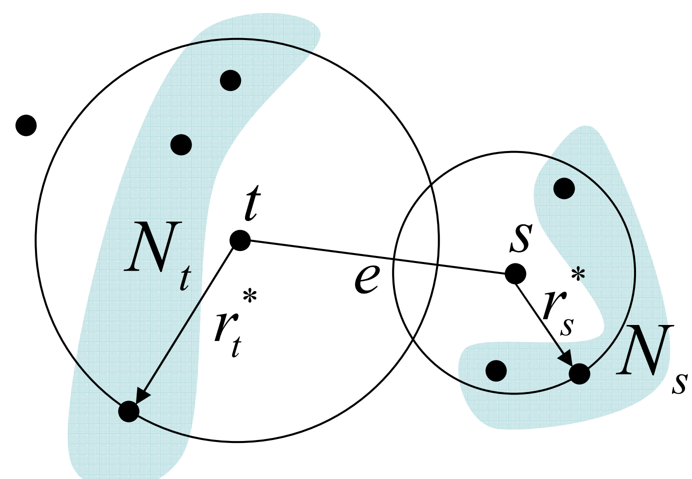
Fault-Tolerant Power Assignment

- **The algorithm**
  - Assign each $t \in T$ the range r_t^* (denote $A_k^{'}$)
  - Compute an MST of T



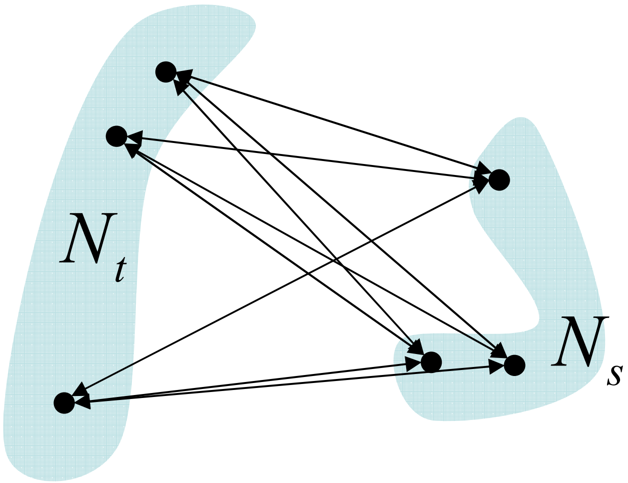
Fault-Tolerant Power Assignment

- **The algorithm**
  - For each edge $e = (t, s)$ of MST increase the range of the nodes in $N_t \cup N_s$ such that each node $t' \in N_t$ can reach all nodes in N_s, and vice versa (denote A_k)



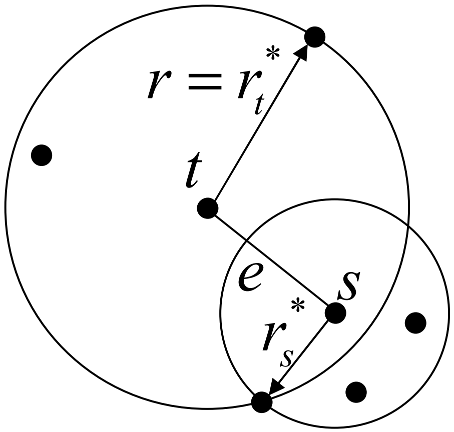
Fault-Tolerant Power Assignment

- **The algorithm**
  - For each edge $e = (t, s)$ of MST increase the range of the nodes in $N_t \cup N_s$ such that each node $t' \in N_t$ can reach all nodes in N_t, and vice versa (denote A_k)



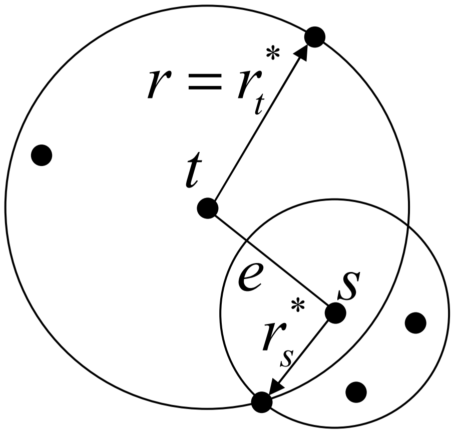
Fault-Tolerant Power Assignment

- **Proof sketch**
  - Let $r = \max\{r_t^*, r_s^*\}$
  - In A_k each $t' \in N_t \cup N_s$ is assigned at most $|e| + 2r$
  - Case 1: $|e| \le r$



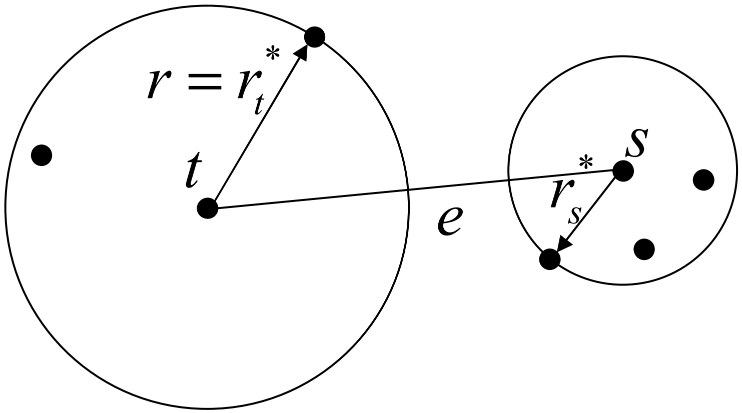
Fault-Tolerant Power Assignment

- **Proof sketch**
  - Let $r = \max\{r_t^*, r_s^*\}$
  - In A_k each $t' \in N_t \cup N_s$ is assigned at most $|e| + 2r$
  - Case 1: $|e| \le r \Rightarrow \max\limits_{t' \in N_t, s' \in N_s} d(t', s') \le 3r$



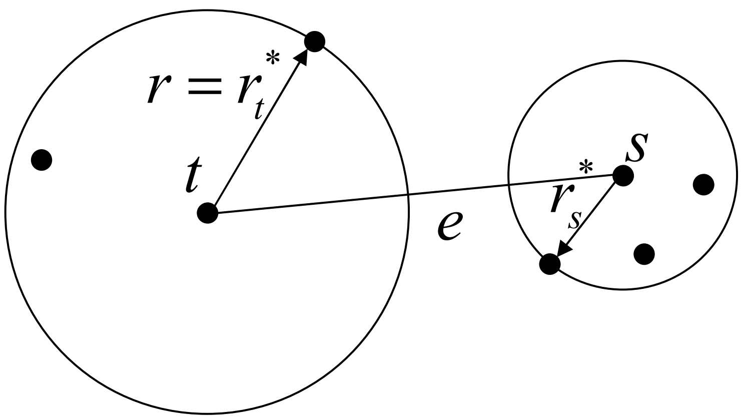
Fault-Tolerant Power Assignment

- **Proof sketch**
  - Let $r = \max\{r_t^*, r_s^*\}$
  - In A_k each $t' \in N_t \cup N_s$ is assigned at most $|e| + 2r$
  - Case 2: $|e| > r$



Fault-Tolerant Power Assignment

- **Proof sketch**
  - Let $r = \max\{r_t^*, r_s^*\}$
  - In A_k each $t' \in N_t \cup N_s$ is assigned at most $|e| + 2r$
  - Case 2: $|e| > r \Rightarrow \max_{t' \in N_t, s' \in N_s} d(t', s') \leq 3|e|$



Fault-Tolerant Power Assignment

- **Proof sketch**
  - Let $r = \max\{r_t^*, r_s^*\}$
  - In A_k each $t' \in N_t \cup N_s$ is assigned at most $|e| + 2r$
  - Easy to see $C_{A_k'} \le C_{A_k^*}$

Fault-Tolerant Power Assignment

- **Proof sketch**
  - Let $r = \max\{r_t^*, r_s^*\}$
  - In A_k each $t' \in N_t \cup N_s$ is assigned at most $|e| + 2r$
  - Easy to see $C_{A_k'} \leq C_{A_k^*}$
  - Kirousis et al. proved $C_{\text{MST}} \leq C_{A_1^*}$

Fault-Tolerant Power Assignment

- **Proof sketch**
  - Let $r = \max\{r_t^*, r_s^*\}$
  - In A_k each $t' \in N_t \cup N_s$ is assigned at most $|e| + 2r$
  - Easy to see $C_{A_k'} \le C_{A_k^*}$
  - Kirousis et al. proved $C_{\mathrm{MST}} \le C_{A_1^*}$
  - As a result

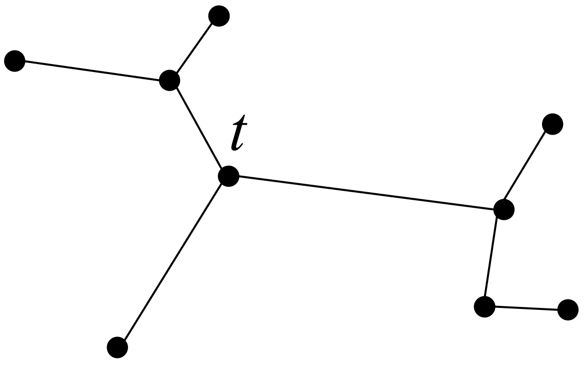
$$C_{A_k} \le O(k) \cdot (C_{A_k^*} + C_{\mathrm{MST}}) \le O(k) \cdot C_{A_k^*}$$

Outline

- Introduction
- Model & Problems
- Fault-Tolerant Power Assignment
- Connected Backbone Power Assignment
- Summary

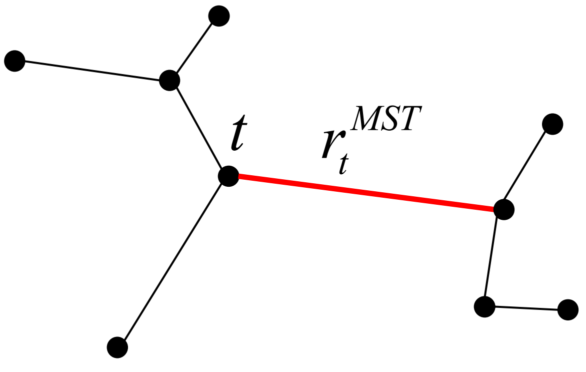
Connected Backbone Power Assignment

- **Definitions**
  - Given the MST of T, for any node $t \in T$, let r_t^{MST} be the size of the longest edge adjacent to t



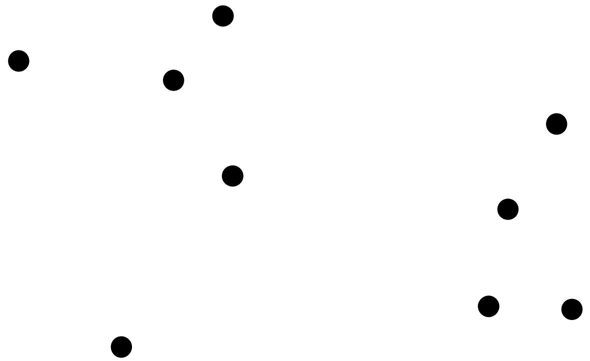
Connected Backbone Power Assignment

- **Definitions**
  - Given the MST of T, for any node $t \in T$, let r_t^{MST} be the size of the longest edge adjacent to t



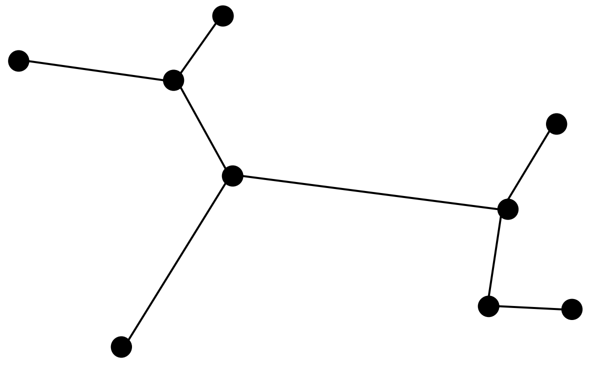
Connected Backbone Power Assignment

- **The algorithm**
  - Compute an MST of *T*



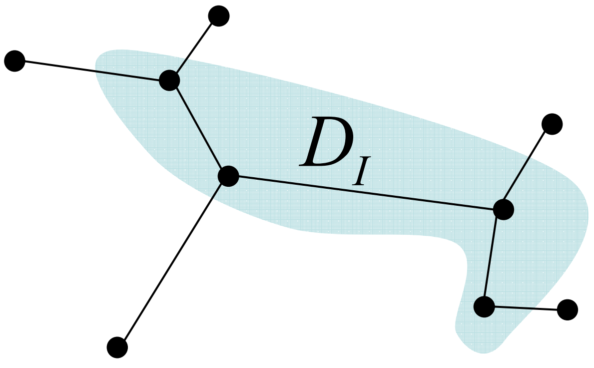
Connected Backbone Power Assignment

- **The algorithm**
  - Compute an MST of T



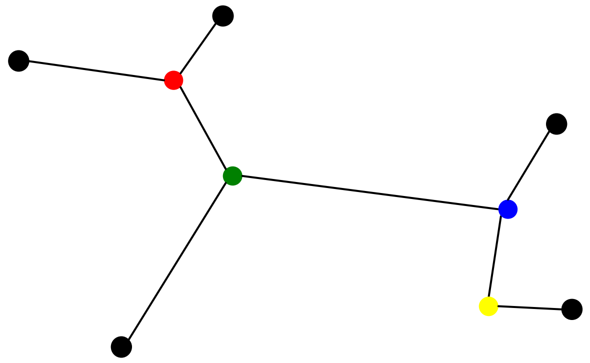
Connected Backbone Power Assignment

- **The algorithm**
  - Compute an MST of T
  - Let D_I be the set of all internal nodes of MST



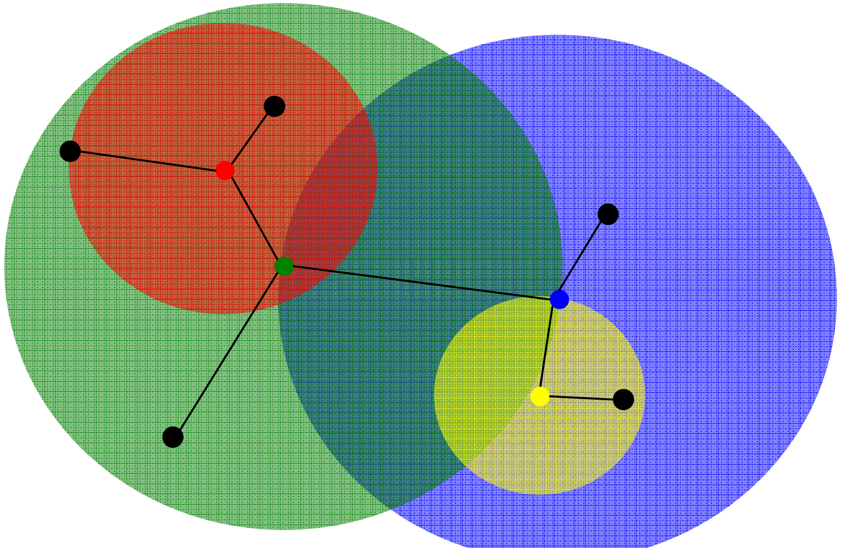
Connected Backbone Power Assignment

- **The algorithm**
  - Compute an MST of T
  - Let D_I be the set of all internal nodes of MST
  - Assign each $u \in D_I$ with r_u^{MST} (denote A)



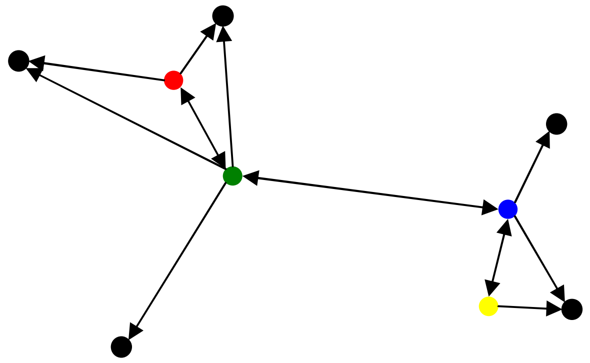
Connected Backbone Power Assignment

- **The algorithm**
  - Compute an MST of T
  - Let D_I be the set of all internal nodes of MST
  - Assign each $u \in D_I$ with r_u^{MST} (denote A)



Connected Backbone Power Assignment

- **The algorithm**
  - Compute an MST of T
  - Let D_I be the set of all internal nodes of MST
  - Assign each $u \in D_I$ with r_u^{MST} (denote A)

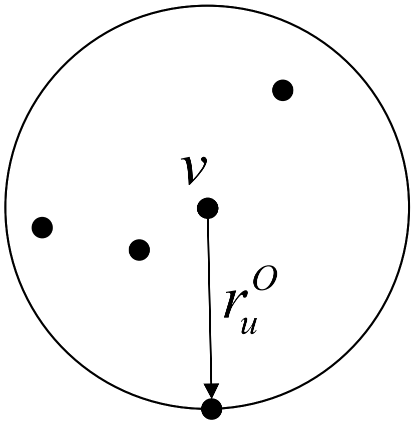


Connected Backbone Power Assignment

- **Proof sketch**
  - Construct a power assignment B for which it holds $C_B \leq c_1 \cdot C_{\mathrm{OPT}}$ and $C_A \leq c_2 \cdot C_B$, as a result obtaining $C_A \leq c \cdot C_{\mathrm{OPT}}$
  - B is derived from OPT

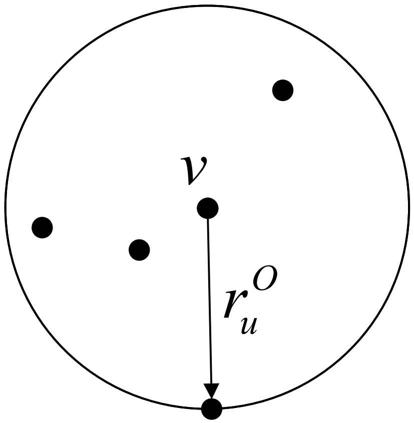
Connected Backbone Power Assignment

- **Proof sketch**
  - Let D_{OPT} be the *connected backbone* in OPT
  - For each node $v \in D_{\mathrm{OPT}}$ let r_v^O be the transmission range of v in OPT



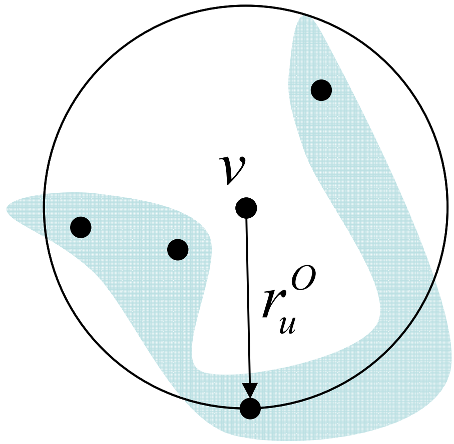
Connected Backbone Power Assignment

- **Proof sketch**
  - For each node $v \in D_{\text{OPT}}$ let T_v be all the nodes within distance r_v^O from v



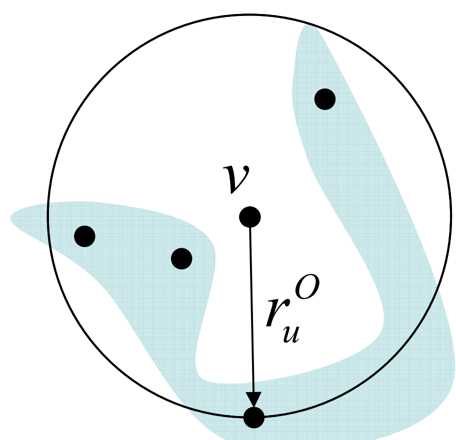
Connected Backbone Power Assignment

- **Proof sketch**
  - For each node $v \in D_{\text{OPT}}$ let T_v be all the nodes within distance r_v^O from v



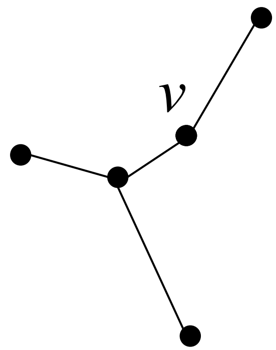
Connected Backbone Power Assignment

- **Proof sketch**
  - For each node $v \in D_{\text{OPT}}$ let T_v be all the nodes within distance r_v^O from v
  - For each node $v \in D_{\text{OPT}}$ compute MST_v of $T_v \cup \{v\}$



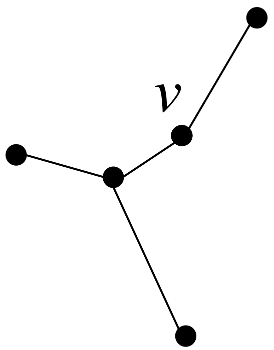
Connected Backbone Power Assignment

- **Proof sketch**
  - For each node $v \in D_{\text{OPT}}$ let T_v be all the nodes within distance r_v^O from v
  - For each node $v \in D_{\text{OPT}}$ compute MST_v of $T_v \cup \{v\}$



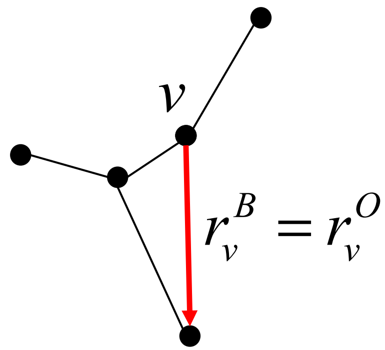
Connected Backbone Power Assignment

- **Proof sketch**
  - In B:
    - Each node $v \in D_{\text{OPT}}$ is assigned $r_v^B = r_v^O$



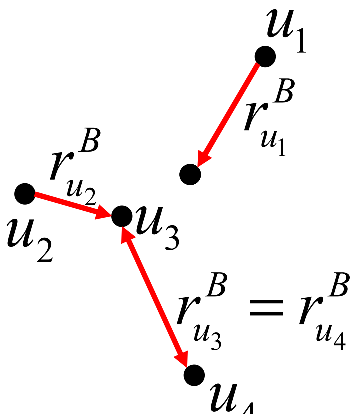
Connected Backbone Power Assignment

- **Proof sketch**
  - In B:
    - Each node $v \in D_{\mathrm{OPT}}$ is assigned $r_v^B = r_v^O$



Connected Backbone Power Assignment

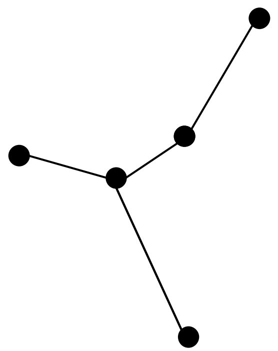
- **Proof sketch**
  - In B:
    - Each node $v \in D_{\text{OPT}}$ is assigned $r_v^B = r_v^O$
    - Each node $u \in T \setminus D_{\text{OPT}}$ is assigned $r_u^B = r_u^{\text{MST}_v}$

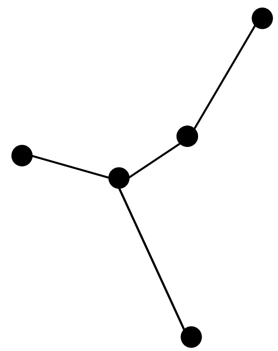


Connected Backbone Power Assignment

- **Proof sketch**
  - $C_B \leq c_1 \cdot C_{\mathrm{OPT}}$
    - Carmi et al. showed that

$$\sum_{e \in \mathrm{MST_v}} area(D_e) \leq 5 area(\cup_{e \in \mathrm{MST_v}} D_e)$$

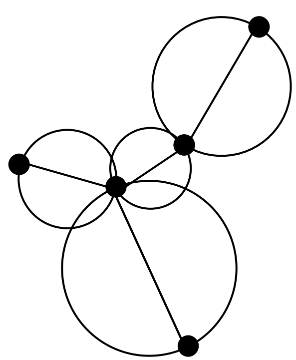


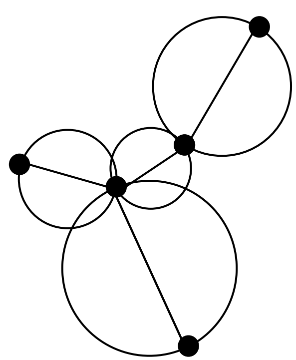


Connected Backbone Power Assignment

- **Proof sketch**
  - $C_B \le c_1 \cdot C_{\mathrm{OPT}}$
    - Carmi et al. showed that

$$\sum_{e \in \mathrm{MST_v}} area(D_e) \le 5 area(\cup_{e \in \mathrm{MST_v}} D_e)$$

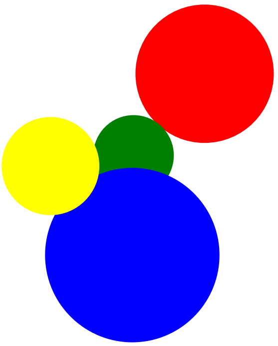


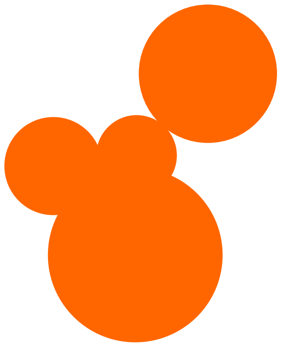


Connected Backbone Power Assignment

- **Proof sketch**
  - $C_B \leq c_1 \cdot C_{\mathrm{OPT}}$
    - Carmi et al. showed that

$$\sum_{e \in \mathrm{MST_v}} area(D_e) \leq 5 area(\cup_{e \in \mathrm{MST_v}} D_e)$$

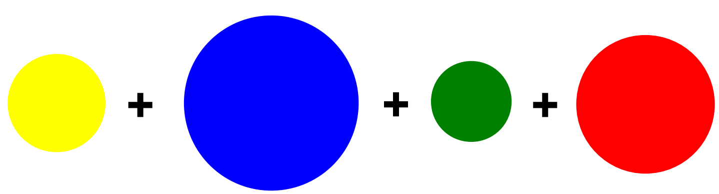




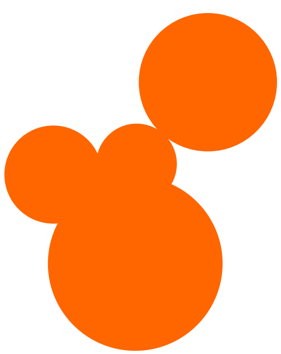
Connected Backbone Power Assignment

- **Proof sketch**
  - $C_B \leq c_1 \cdot C_{\mathrm{OPT}}$
    - Carmi et al. showed that

$$\sum_{e\in \mathrm{MST_v}} area(D_e) \leq 5area(\cup_{e\in \mathrm{MST_v}} D_e)$$



≤ 5



Connected Backbone Power Assignment

- **Proof sketch**
  - $C_B \le c_1 \cdot C_{\mathrm{OPT}}$
    - Carmi et al. showed that

$$\sum_{e \in \mathrm{MST_v}} area(D_e) \le 5area(\cup_{e \in \mathrm{MST_v}} D_e)$$

    - Using this fact we obtain

$$C_B = O(C_{\mathrm{OPT}})$$

Connected Backbone Power Assignment

- **Proof sketch**
  - $C_A \le c_2 \cdot C_B$
    - Kirousis et al. proved that given an MST assigning each node $v \in T$ with r_v^{MST} yields a 2-factor approximation for *strong-connectivity* (denote A_{SC})

Connected Backbone Power Assignment

- **Proof sketch**
  - $C_A \le c_2 \cdot C_B$
    - Kirousis et al. proved that given an MST assigning each node $v \in T$ with r_v^{MST} yields a 2-factor approximation for *strong-connectivity* (denote A_{SC})
  - Using this fact we obtain

$$C_A \le C_{A_{SC}} \le 2C_{A_1^*} \le 2C_B$$

Connected Backbone Power Assignment

- **Proof sketch**
  - Therefore

$$C_A = O(C_{\mathrm{OPT}})$$

Outline

- Introduction
- Model & Problems
- Fault-Tolerant Power Assignment
- Connected Backbone Power Assignment
- Summary

Summary

- **Problem 1** (k-vertex-connectivity)

  Input: A set T of transceivers, and a parameter $k > 1$

  Output: A power assignment $A(T)$ with minimal possible cost C_A, where H_A is *k-vertex connected*

$O(k)$**-approximation algorithm**

Summary

- **Problem 2** (connected backbone)

  Input: A set T of transceivers

  Output: A subset D of T and a power assignment $A(D)$ with minimal possible cost C_A, where H_A (restricted to D) is *strongly connected*, and for each $t \in T \setminus D$, there exists $u \in D$, such that $d(u,t) \le r_u$

Constant-factor approximation algorithm in $O(n \log n)$

Summary

- **Problem 3** (k-connected backbone)

  Input: A set T of transceivers, and a parameter $k > 1$

  Output: A subset D of T and a power assignment $A(D)$ with minimal possible cost C_A, where H_A (restricted to D) is *k-vertex connected*, and for each $t \in T \setminus D$, there exists $u_1, u_2, \ldots, u_k \in D$, such that $d(u_i, t) \le r_{u_i}$

$O(k^3)$**-approximation algorithm**

?