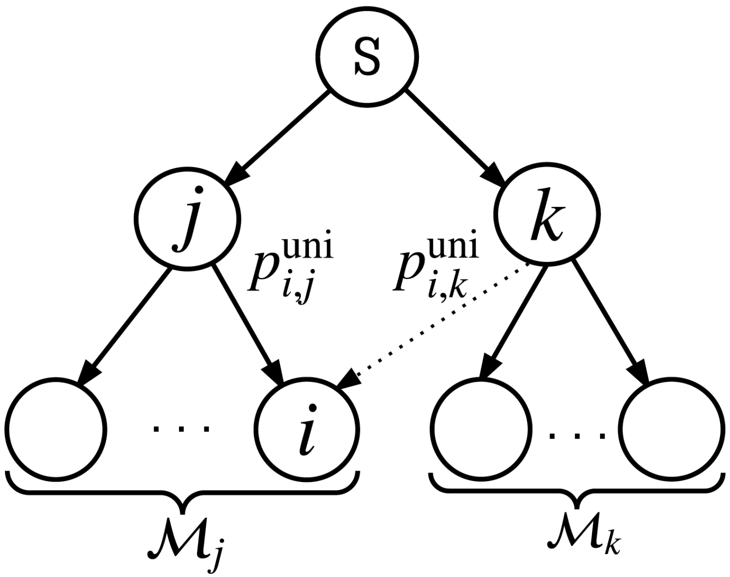
Energy-Efficient Broadcast Tree Construction Protocol for Wireless Multi-Hop Networks

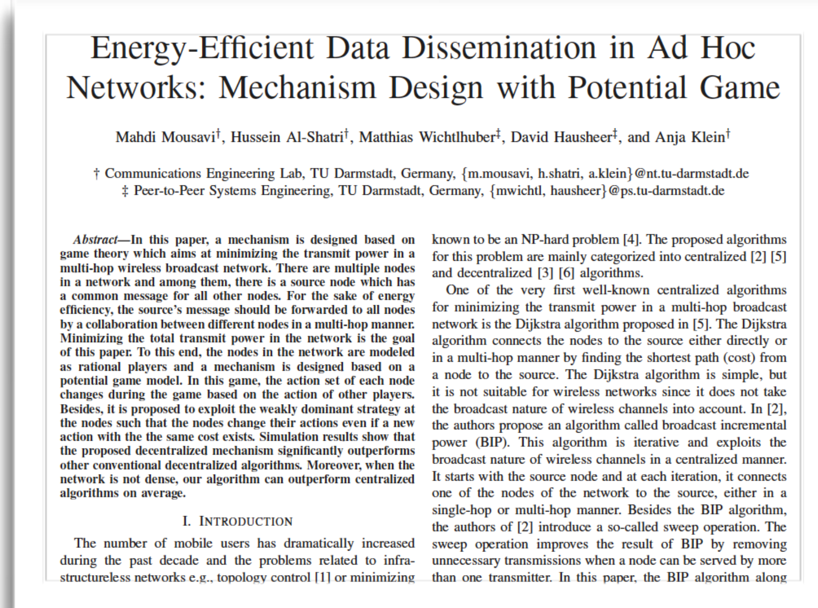
MAKI Seminar 21st Jan. 2021

C1 Phase II: Robin Klose, Kevin Küchler
B3 Phase II: Mahdi Mousavi

TECHNISCHE UNIVERSITÄT DARMSTADT

SECURE MOBILE NETWORKING





Energy-Efficient Data Dissemination in Ad Hoc Networks: Mechanism Design with Potential Game

Mahdi Mousavi[†], Hussein Al-Shatri[†], Matthias Wichtlhuber[‡], David Hausheer[‡], and Anja Klein[†]

† Communications Engineering Lab, TU Darmstadt, Germany, {m.mousavi, h.shatri, a.klein}@nt.tu-darmstadt.de
‡ Peer-to-Peer Systems Engineering, TU Darmstadt, Germany, {mwichtl, hausheer}@ps.tu-darmstadt.de

***Abstract*—In this paper, a mechanism is designed based on game theory which aims at minimizing the transmit power in a multi-hop wireless broadcast network. There are multiple nodes in a network and among them, there is a source node which has a common message for all other nodes. For the sake of energy efficiency, the source's message should be forwarded to all nodes by a collaboration between different nodes in a multi-hop manner. Minimizing the total transmit power in the network is the goal of this paper. To this end, the nodes in the network are modeled as rational players and a mechanism is designed based on a potential game model. In this game, the action set of each node changes during the game based on the action of other players. Besides, it is proposed to exploit the weakly dominant strategy at the nodes such that the nodes change their actions even if a new action with the the same cost exists. Simulation results show that the proposed decentralized mechanism significantly outperforms other conventional decentralized algorithms. Moreover, when the network is not dense, our algorithm can outperform centralized algorithms on average.**

I. INTRODUCTION

The number of mobile users has dramatically increased during the past decade and the problems related to infrastructureless networks e.g., topology control [1] or minimizing

known to be an NP-hard problem [4]. The proposed algorithms for this problem are mainly categorized into centralized [2] [5] and decentralized [3] [6] algorithms.

One of the very first well-known centralized algorithms for minimizing the transmit power in a multi-hop broadcast network is the Dijkstra algorithm proposed in [5]. The Dijkstra algorithm connects the nodes to the source either directly or in a multi-hop manner by finding the shortest path (cost) from a node to the source. The Dijkstra algorithm is simple, but it is not suitable for wireless networks since it does not take the broadcast nature of wireless channels into account. In [2], the authors propose an algorithm called broadcast incremental power (BIP). This algorithm is iterative and exploits the broadcast nature of wireless channels in a centralized manner. It starts with the source node and at each iteration, it connects one of the nodes of the network to the source, either in a single-hop or multi-hop manner. Besides the BIP algorithm, the authors of [2] introduce a so-called sweep operation. The sweep operation improves the result of BIP by removing unnecessary transmissions when a node can be served by more than one transmitter. In this paper, the BIP algorithm along

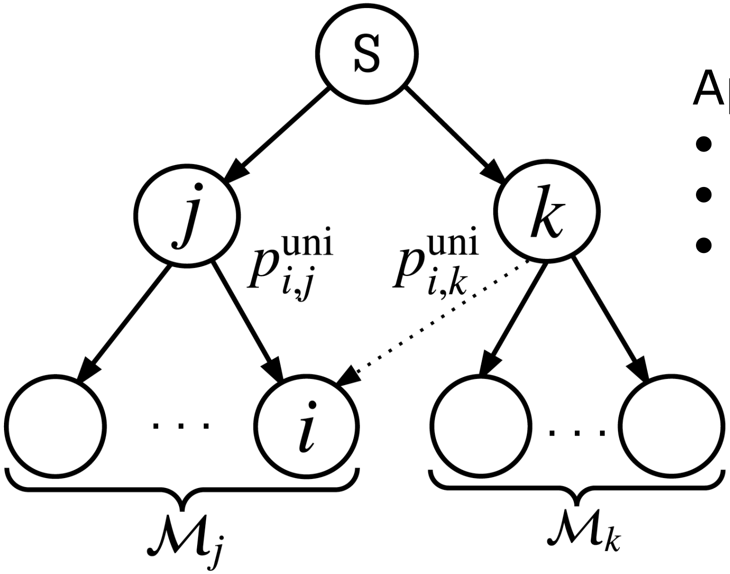
DFG Collaborative Research Centre 1053 – MAKI

Introduction

Scenario: broadcasting in wireless multi-hop network

- Source node S has data for all other nodes.
- Nodes form a multi-hop wireless broadcast network.

Goal: Minimize the total transmission power in the network.



Approach by Mahdi Mousavi et al. [1]:

- Model the nodes as rational players
- Apply a potential game model
- Nodes ...
  - connect to a parent node in the tree
  - as a function of other nodes' actions
  - while keeping the imposed cost minimal
  - by utilizing wireless broadcast characteristics.

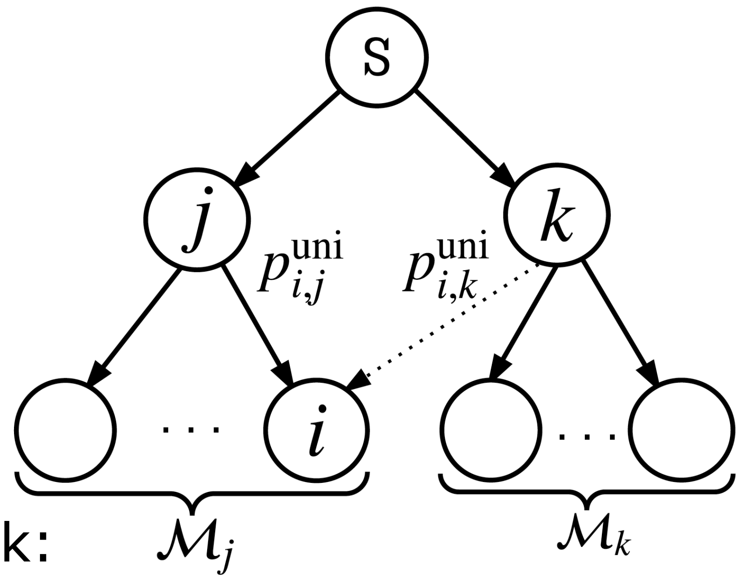
Potential Game Model

SECURE MOBILE NETWORKING

Required transmission power for j → i: $p_{i,j}$

Transmission power of parent node j:

$$p_j^{\text{tx}}(\mathcal{M}_j) = \max_{i \in \mathcal{M}_j} \{p_{i,j}^{\text{uni}}\}$$



Total transmission power in the network:

$$p^{\text{net}} = \sum_{j=1}^{N+1} p_j^{\text{tx}}$$

TECHNISCHE UNIVERSITÄT DARMSTADT

Potential Game Model

A node can select a parent node

- that is connected to the broadcast tree
- that does not have this node on its path to the source.

→ **Prevent cycles!**

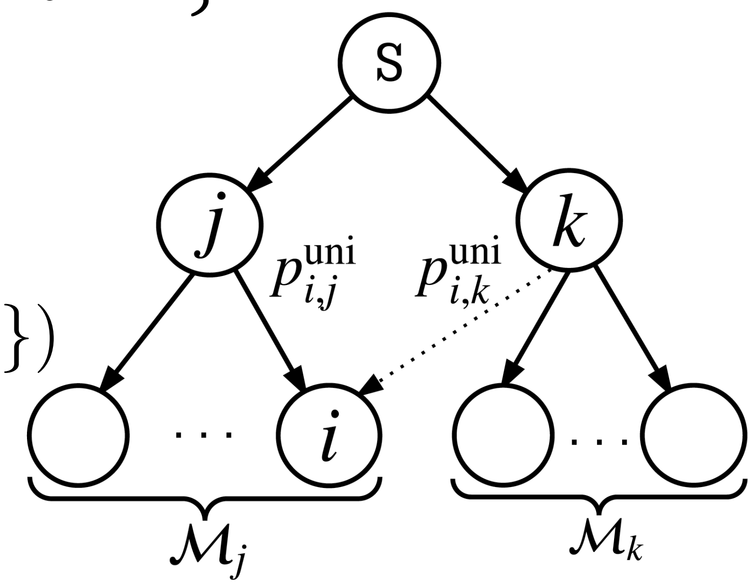
$$\mathcal{A}_i^{(t)} = \left\{ j | j \in \mathcal{N}_i, \mathcal{R}_j^{(t-1)} \neq \{\}, i \notin \mathcal{R}_j^{(t-1)} \right\}$$

Each node minimizes its own cost

- Incentive to select common parents
- Marginal contribution principle

$$\mathcal{C}_i(j, a_{-i}) = p_j^{\text{tx}}(\mathcal{M}_j) - p_j^{\text{tx}}(\mathcal{M}_j \setminus \{i\})$$

$$a_i = \underset{a_i \in \mathcal{A}_i^{(t)}}{\text{argmin}}\, \mathcal{C}_i(a_i, a_{-i}), \forall i \in \mathcal{N}$$



Potential Game Model

Broadcast tree construction phase

- Weakly dominant strategy
  - Nodes may change their parents also after initial connection
  - New cost ≤ old cost
- A new parent must be connected to the broadcast tree
- A connection to a new parent must not create a cycle
- A node finishes its game after N rounds without changes
- The algorithm terminates when all nodes have finished

Applicability in Practical Networks

Advantages of the algorithm by Mahdi Mousavi et al. [1]:

- No central unit for decision-making required
- Better performance than previous works

Deficits: (still some) idealized assumptions

- Global view of the network
    - Knowledge about actions of all other nodes
    - Knowledge of potential parent nodes

$$\mathcal{N}_i = \{j | j \in \mathcal{P} \cup \{S\}, p_{i,j}^{\mathrm{uni}} \leq p^{\mathrm{max}}\}$$

- Discrete time steps (iterations), i.e., execution is "clocked"

Practical Protocol Design: Goals

Adress idealized assumptions

- Discover neighborhood
- Learn about relevant actions of other nodes
- Define sequencing of fine-granular protocol steps
- Deal with packet loss
    - Packet collisions
    - Large-scale interference
- Prevent or resolve cycles
- Adress implementation-specific issues

Practical Protocol Design: Research Questions

- Break-even point: After what amount of data or time does the construction of an energy-efficient broadcast tree pay off?
  - Energy overhead
  - Time overhead
- Compare different design variants under different conditions
  - Connectivity of nodes
  - Packet reception rates
  - Large-scale interference and packet collisions

Energy-Efficient Broadcast Tree Protocol (EEBTP)

Key aspects of EEBTP → M.Sc. thesis by Kevin Küchler [2]

- Apply algorithm by Mahdi Mousavi et al. [1]
- Implementation and integration in ns-3
- Cycle prevention or resolution
    - Ping-to-source
    - Mutex
    - Path-to-source
- Custom energy model
    - Combine energy curves of MAX2828 and MAX2831
- Conventional flooding as a baseline for comparison

EEBTP: Frame Types

- Neighbor discovery
- Child request
- Child confirmation
- Child rejection
- Parent revocation
- Cycle detection/prevention
- End of game
- Application data

EEBTP Cycle Detection: Ping-to-Source

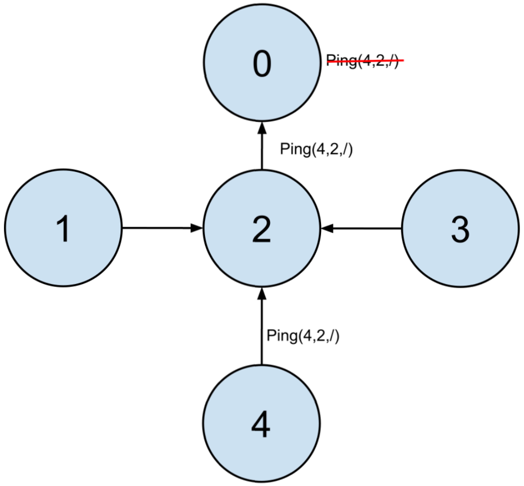
Send a "ping" message to the source node via new parent

- Message contains
  - Sender
  - Old parent node
  - New parent node
- If message arrives at its sender
  - Cycle is detected
  - Connect to old parent
- Nodes maintain a blacklist and a stack of old parents
  - Avoid continuously switching back and forth

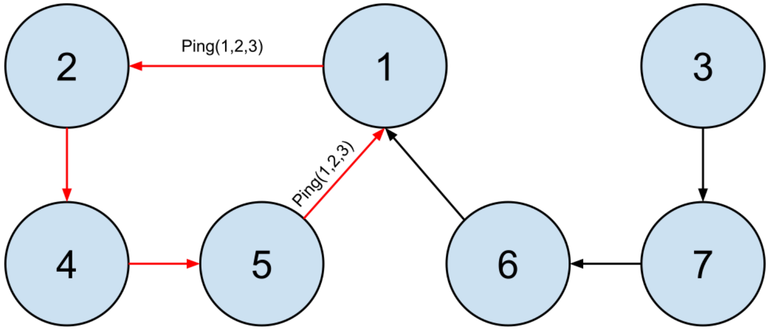
Best-effort approach

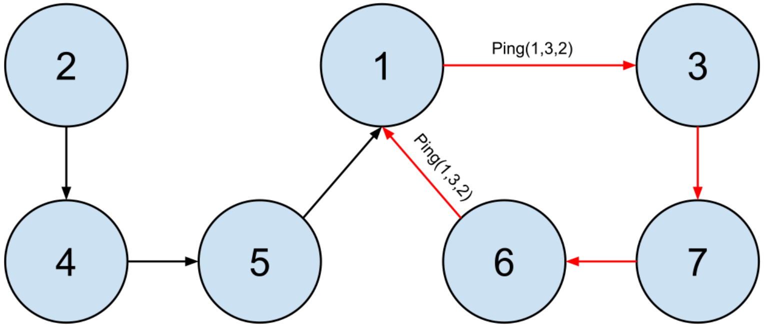
- Save time by asynchronous operation
- Information may get lost
- Cycles can occur (and be resolved if detected)

EEBTP Cycle Detection: Ping-to-Source



(originator / new parent / old parent)





EEBTP Cycle Prevention: Mutex

When attempting to connect to a new parent node

- Lock all child nodes in subtree
- Connect to new parent node
- Release lock

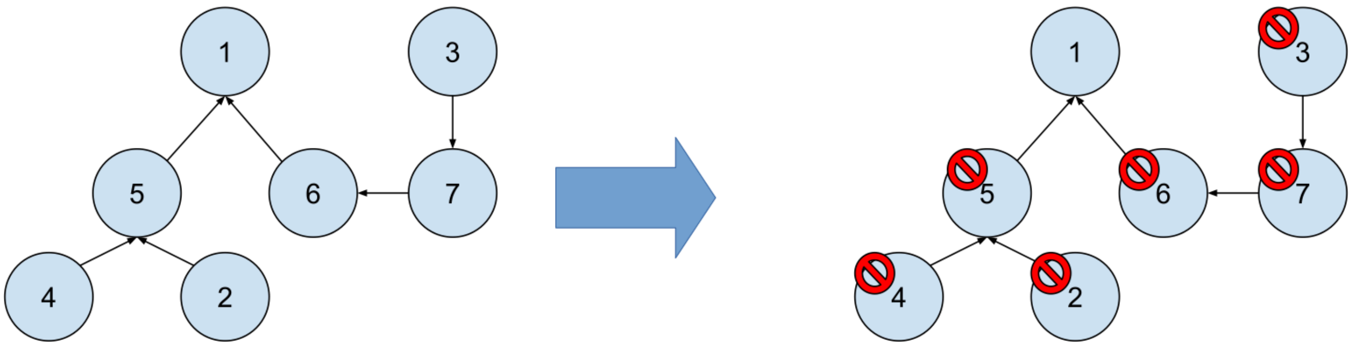
Mutex prevents changes in a subtree

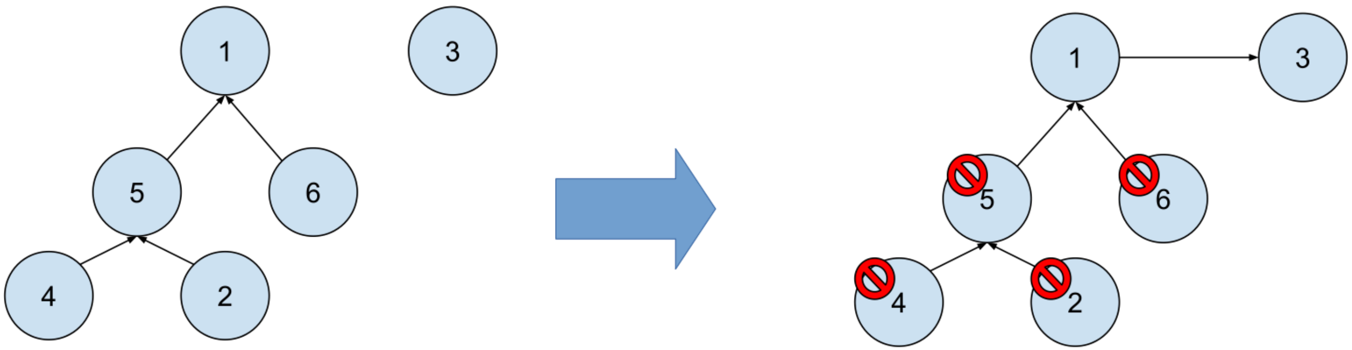
- Mutex request propagates down the subtree
- Confirmations propagate up to the lock holder
- If new parent is locked, it must reject child requests

Effective but high overhead

  - Many messages
  - Takes a lot of time

EEBTP Cycle Prevention: Mutex





EEBTP Cycle Prevention: Path-to-Source

Each node maintains its path to the source node

- Path information included in every neighbor discovery frame
- Receiver checks if it is on the path of the neighbor
- If not, it may connect to the neighbor

Best-effort approach

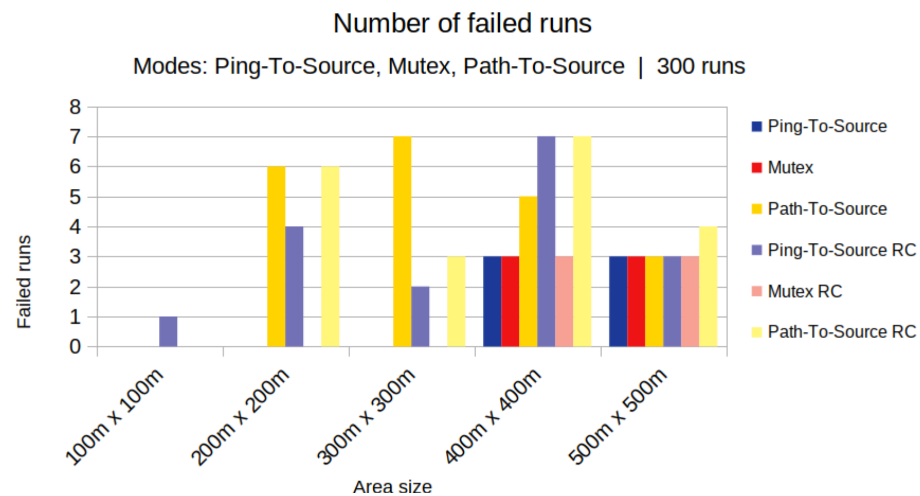
- Save time by asynchronous operation
- Information may get lost
- Cycles can occur (and be resolved if detected)

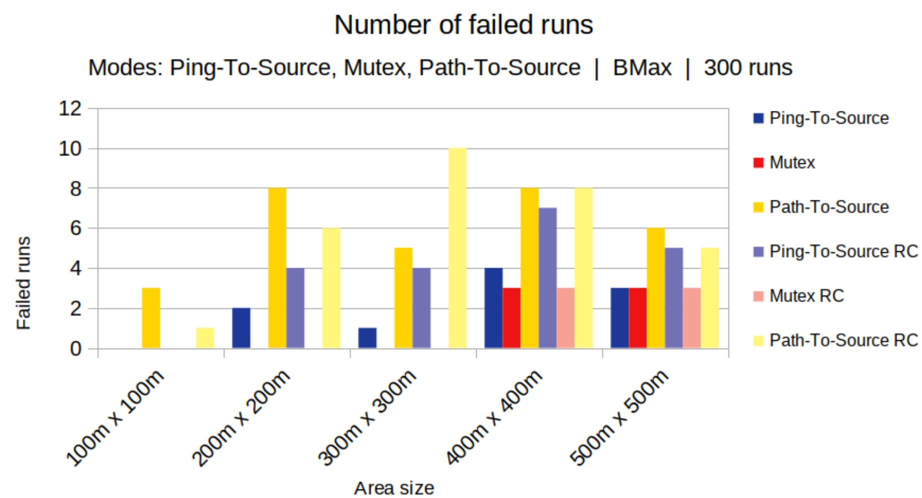
Evaluation

- Failed runs
- Comparison of cycle prevention/detection mechanisms
  - Energy to build a broadcast tree
  - Accumulated TX power
  - Time to build a broadcast tree
  - Tree depth and packet loss
  - Energy to transfer 1MB of data
  - Unconnected nodes
- Conventional flooding with *"Simple Broadcast"*
- Request-To-Send and Clear-To-Send (RTS/CTS; RC)
- Construct tree with max TX power

Evaluation: Failed Runs

- Build time :=
  finish time of source node
- Build phase takes too long
  (more than 10 seconds)
- Build phase never completes
  - A node continuously switches between the same parent nodes
  - Area too large
  - Isolation of nodes

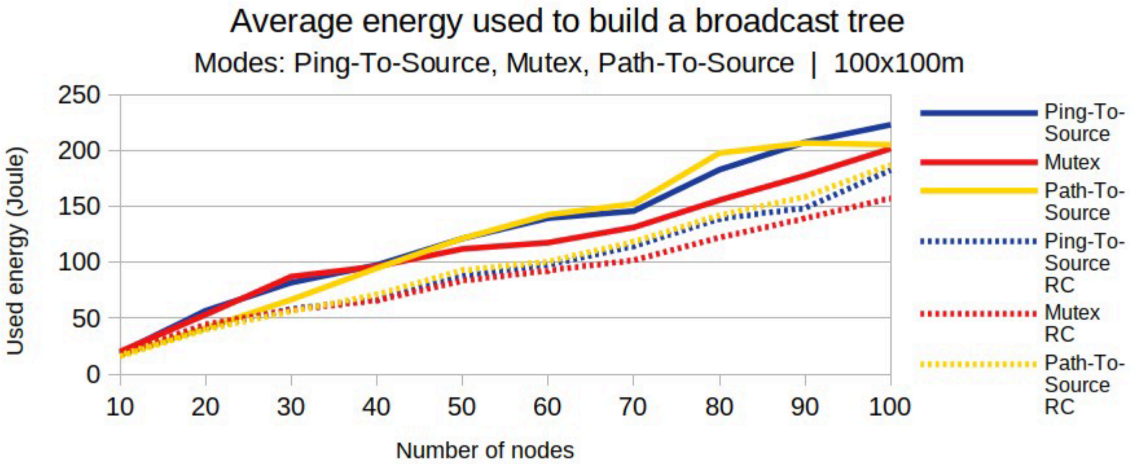


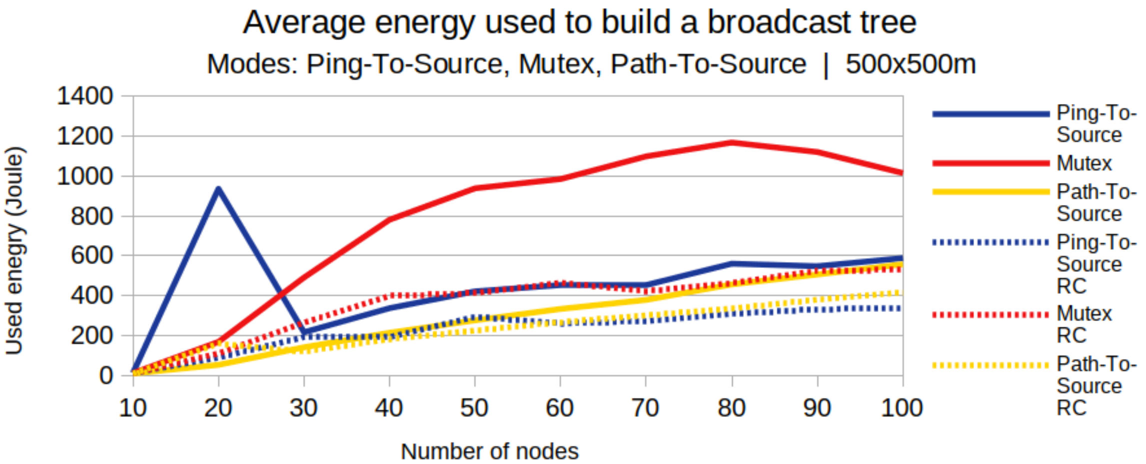


Evaluation: Construction Energy

- *Mutex* needs more energy due to lock request/response overhead. Child requests are rejected when node is locked.

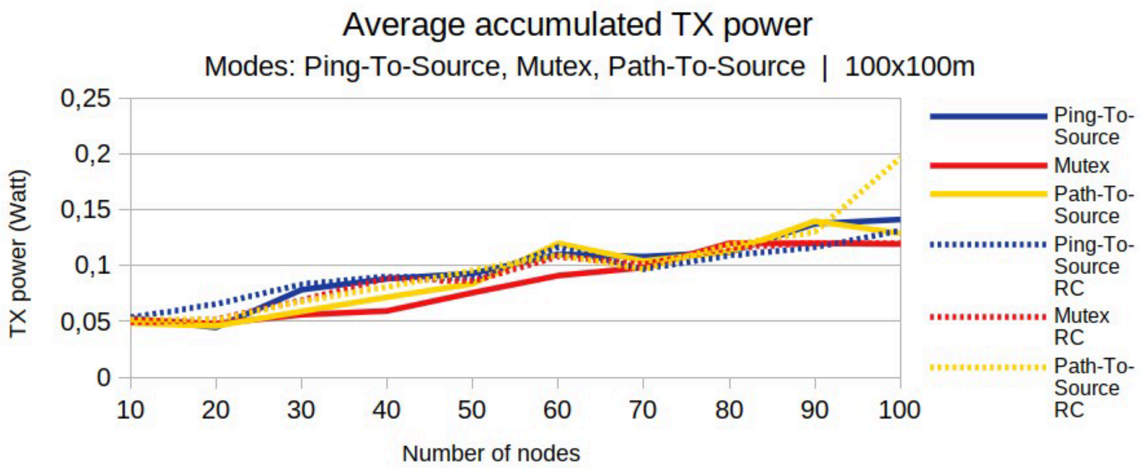
- RTS/CTS: helps to reduce the energy to build the broadcast tree.



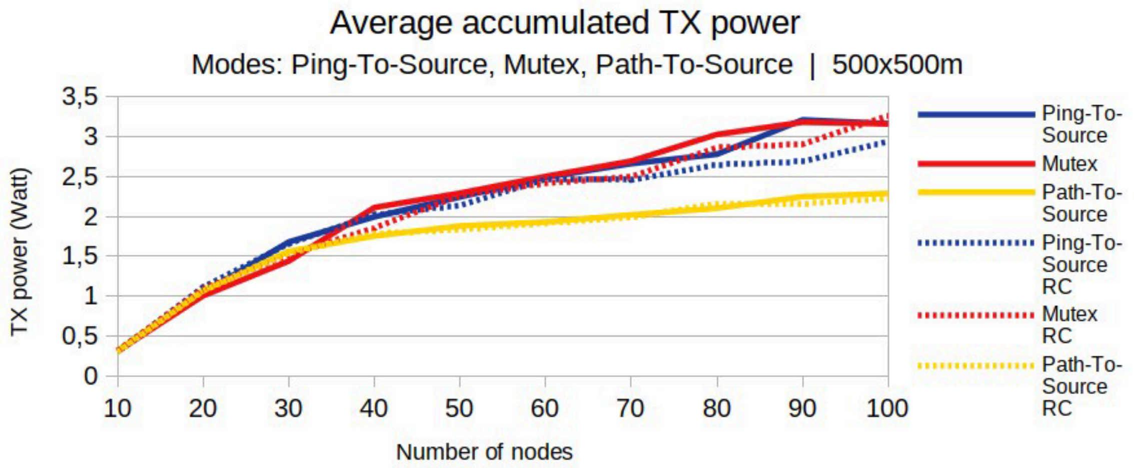


Evaluation: Accumulated TX Power

- Results are similar in small networks and in dense areas.

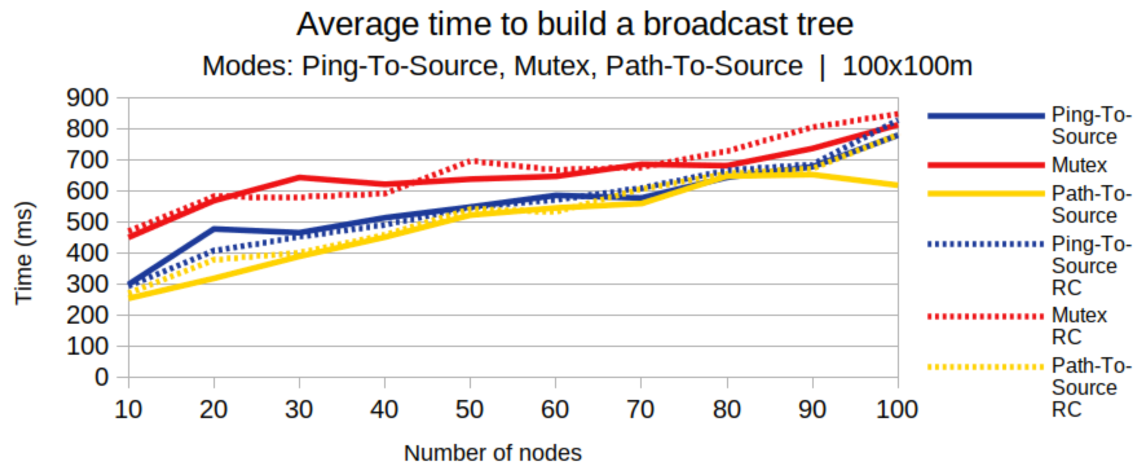


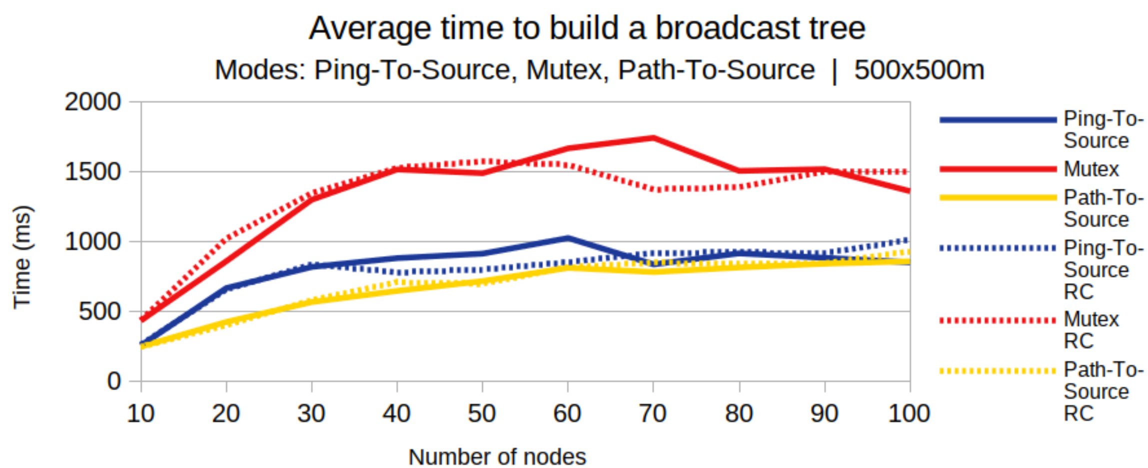
- *Path-to-Source* scores better in large and low density areas but it can create cycles.



Evaluation: Build Time

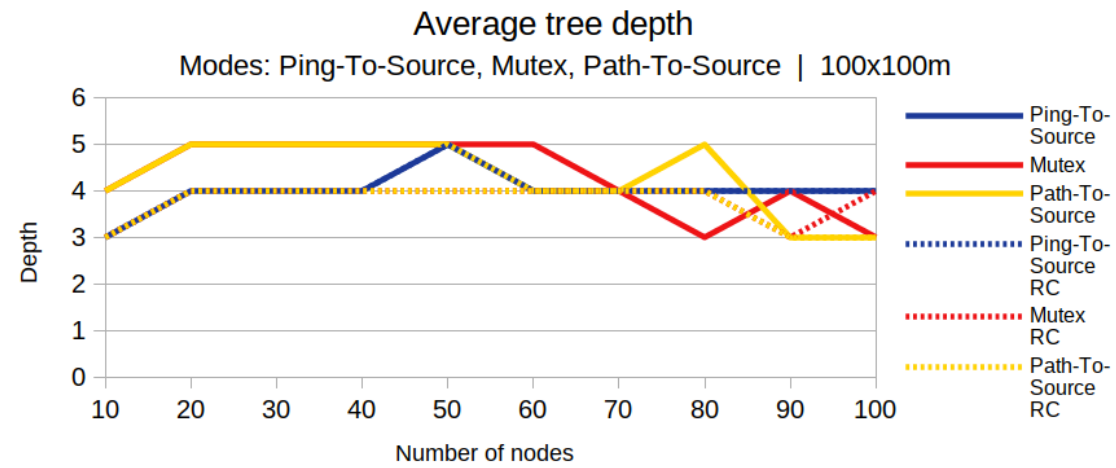
- Build time increases with node count and area size.
- *Mutex* takes longer due to lock requests and responses.
- RTS/CTS has a negligible effect.

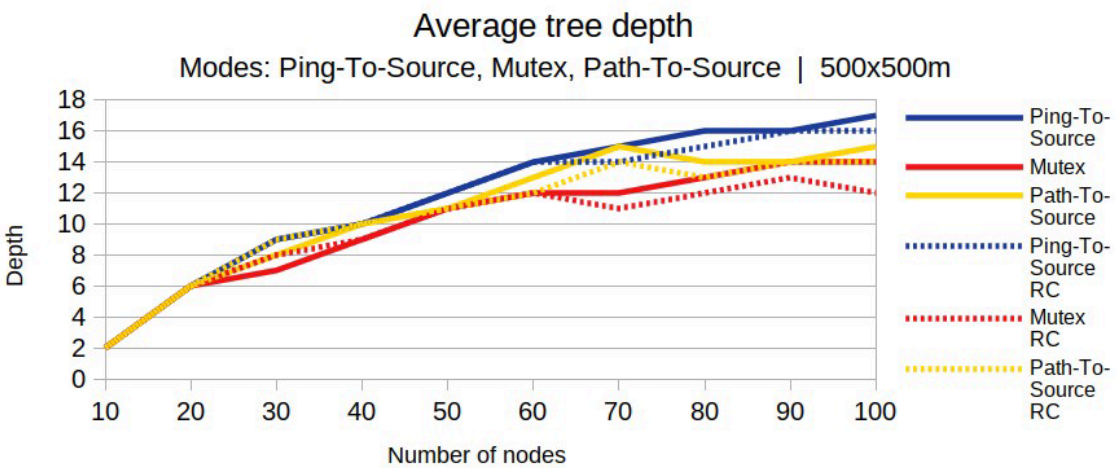




Evaluation: Tree Depth

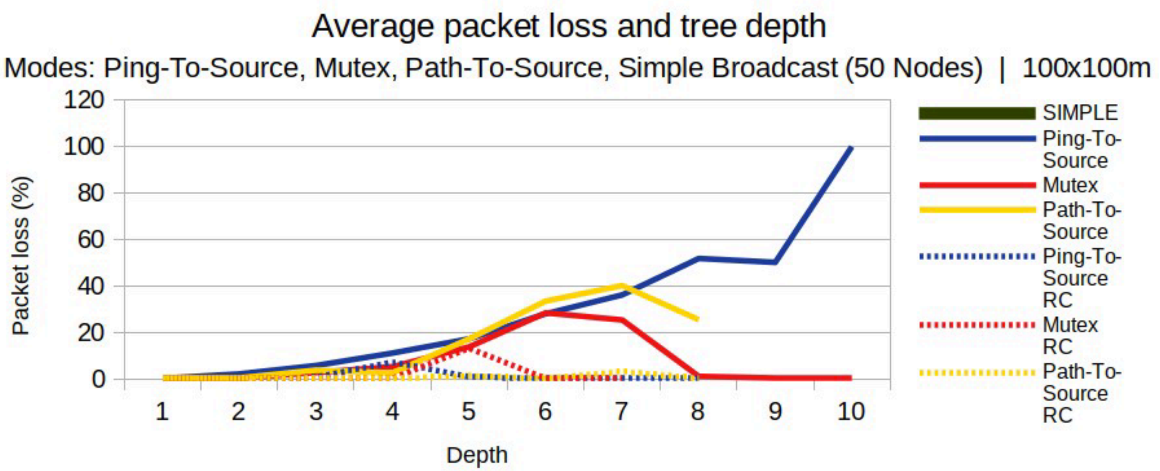
- Small, dense areas result in flat trees.
- Large and sparse areas result in large trees.
- *Ping-to-Source* creates deeper trees than *Mutex*.
- RTS/CTS has a negligible effect.



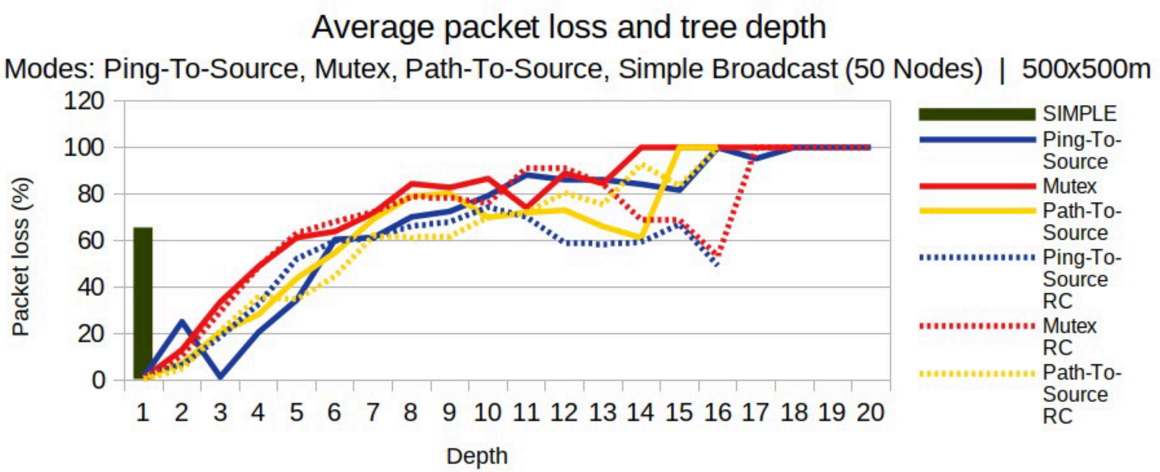


Evaluation: Tree Depth and Packet Loss

- Packet loss increases with tree depth.
- Accepting packet from everyone and not only the parent helps.



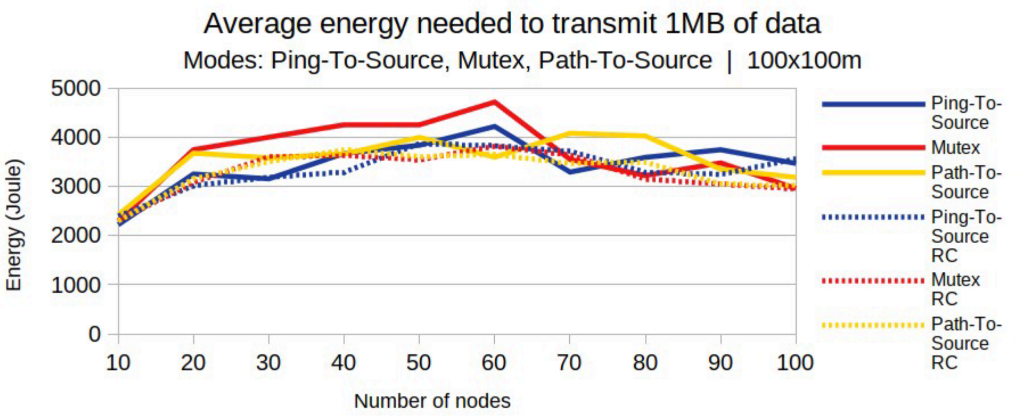
- Packet loss increases in large areas.
- EEBTP may have lower packet loss than *Simple Broadcast*.



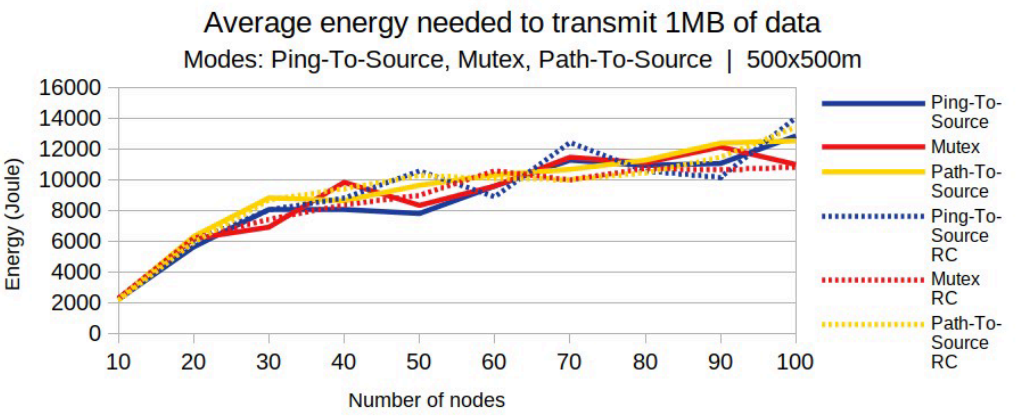
TECHNISCHE UNIVERSITÄT DARMSTADT

Evaluation: TX Energy

- Differences between mechanisms are small in large areas.



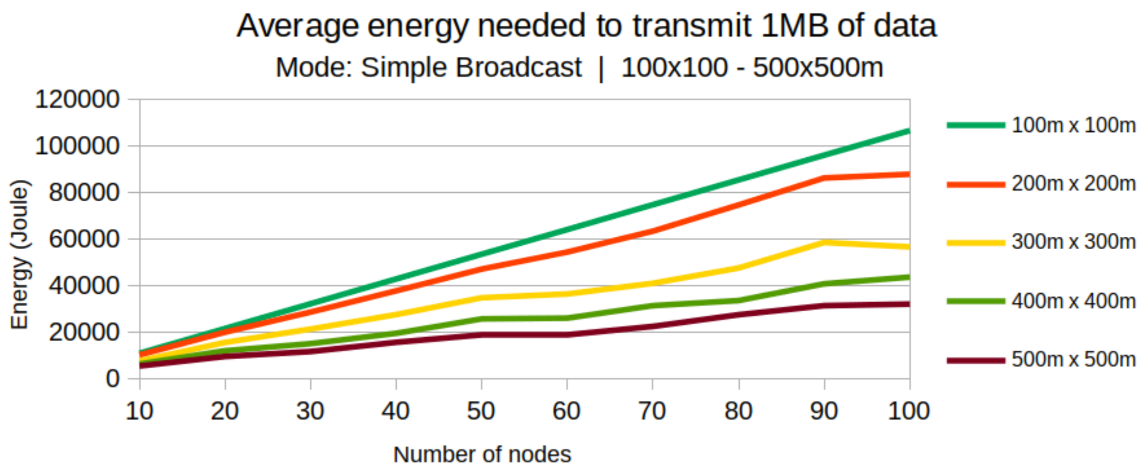
- In large areas more energy is required with an increasing number of nodes.



Evaluation: TX Energy

- *Simple Broadcast* needs up to ~25x more energy.

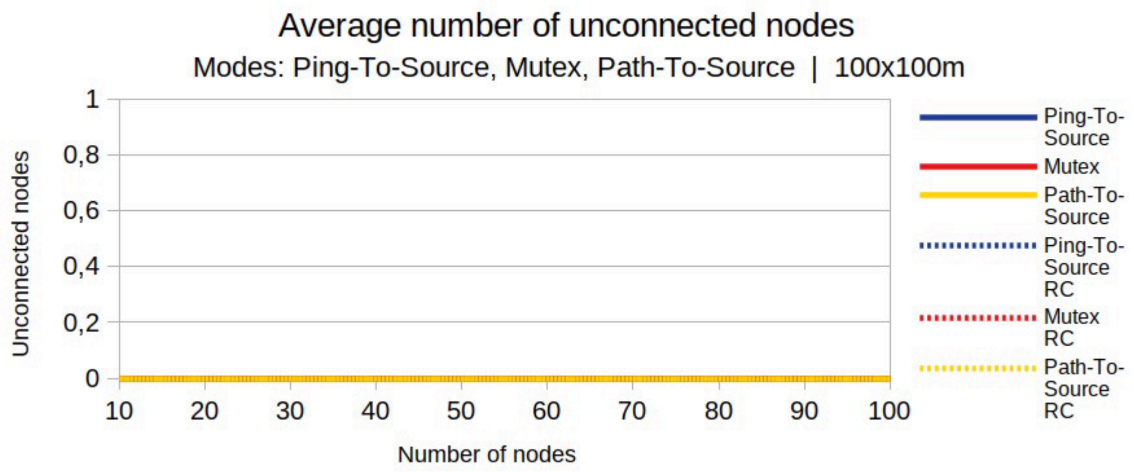
- In larger areas, energy consumption gets lower due to higher packet loss.

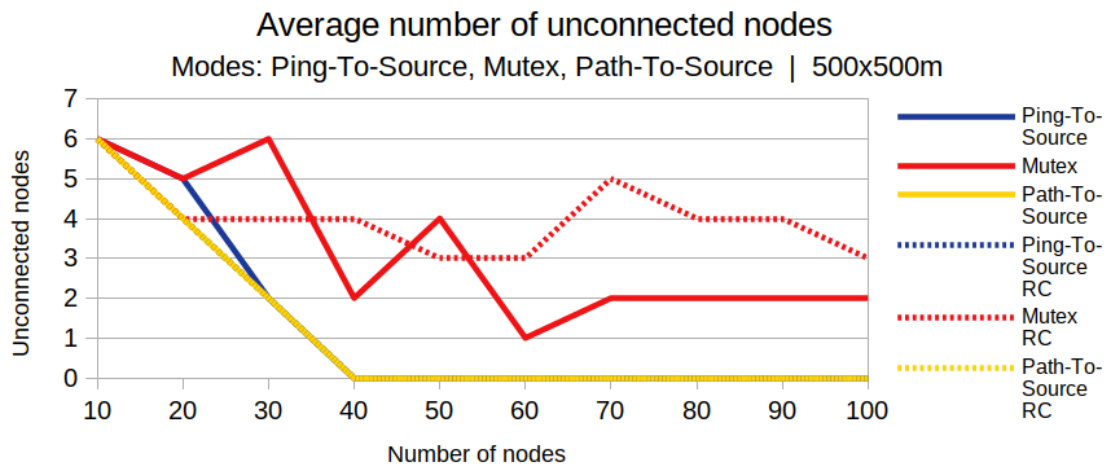


Evaluation: Unconnected Nodes

SECURE MOBILE NETWORKING

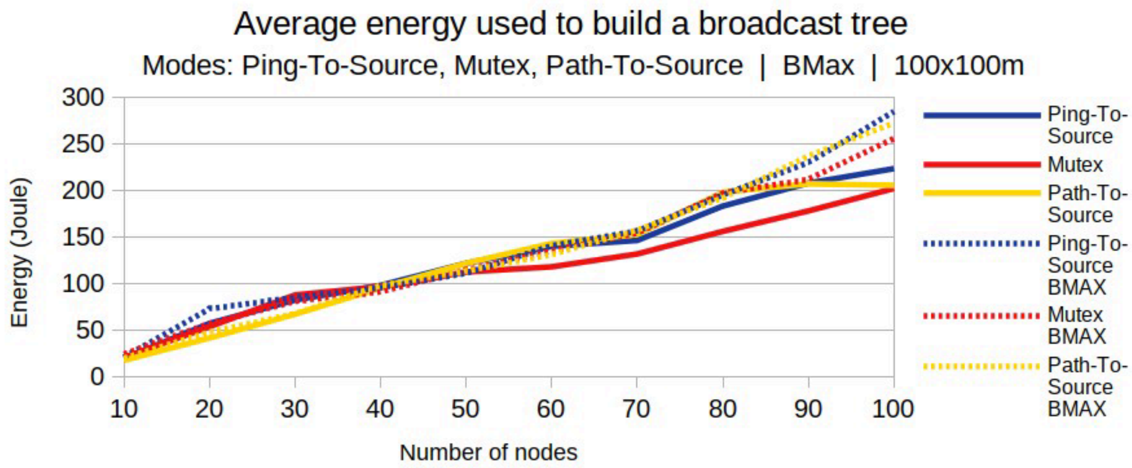
- In large areas, nodes can be too far away to reach each other.
- *Mutex* has more unconnected nodes due to high block times.

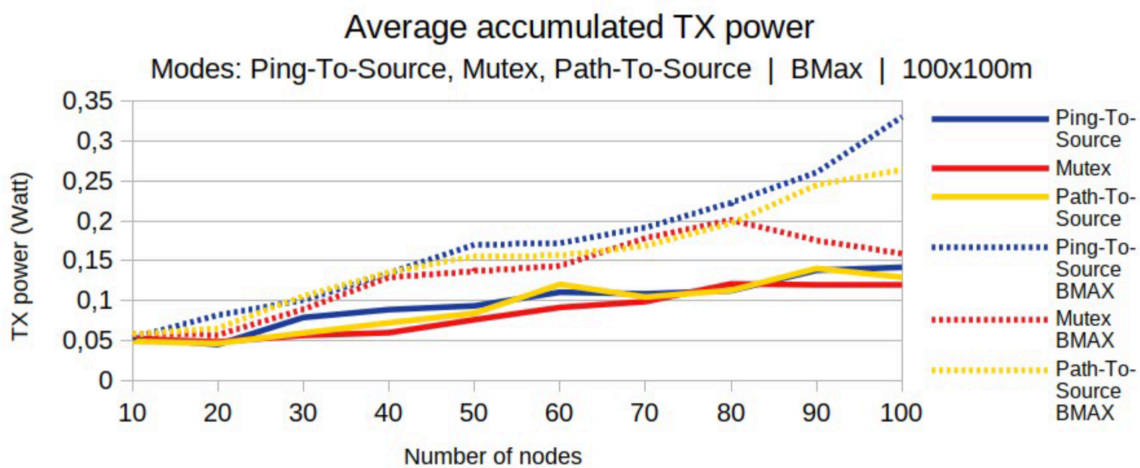




Evaluation: Build with max TX Power

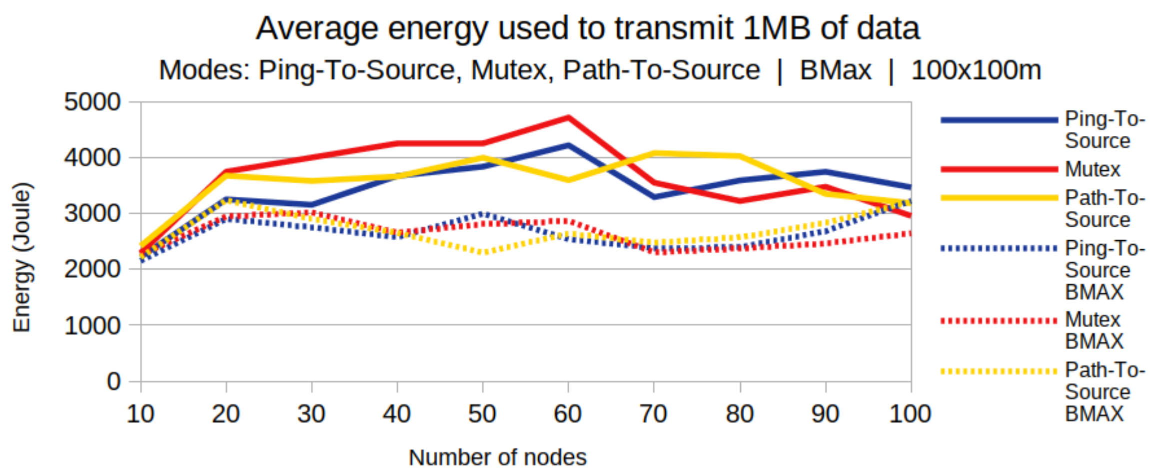
- *BMax* uses the maximum TX power only during the build phase.
- *BMax* needs slightly more energy to build a broadcast tree.
- The accumulated TX power is much higher.
- The build times are nearly identical.

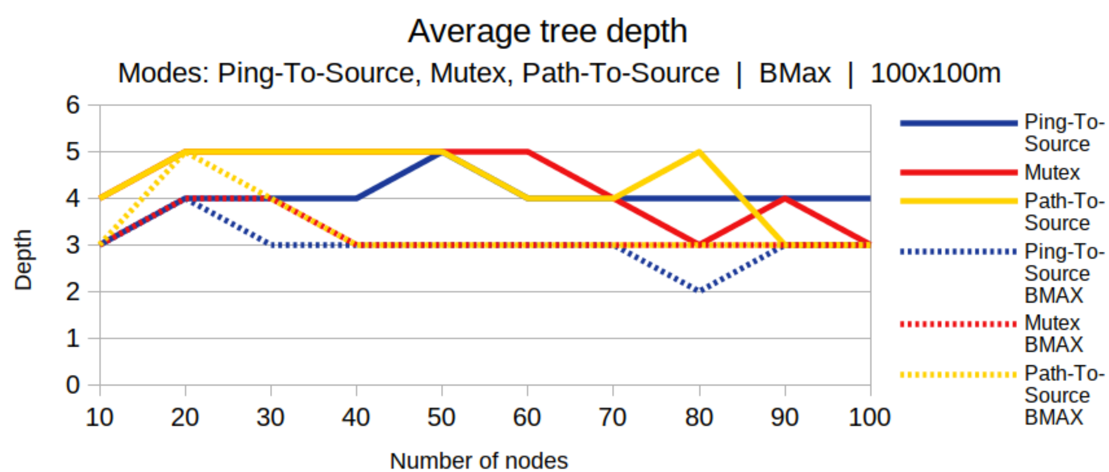




Evaluation: Build with max TX Power

- With *BMax*, the broadcast tree needs less energy to transfer 1 MB of data.
- The average tree depth is lower with *BMax*.
- Packet loss is similar.
- Effect of *BMax* reduces for larger areas.





Discussion

- EEBTP design goal reached: energy reduction for data transmission.
- EEBTP suffers from high noise and interference levels.
- *Ping-To-Source* appears to be most suitable for cycle detection or prevention.
- Mutex does not create cycles but needs more time and build energy.
- Path-To-Source is not suitable since cycles can occur due to lost information.
- Nodes in a cycle can still receive data when they also accept data from nodes other than their parents.
- Large trees not suitable for VoIP or video streaming due to high packet loss.
- RTS/CTS helps to reduce energy during the build phase and the data transmission phase.
- Using the max TX power during the build phase (*BMax*) helps to decrease the needed energy for data transmissions.

Questions and Answers

Thanks for your attention!

Q! & A?

[1] Mahdi Mousavi, Hussein Al-Shatri, Matthias Wichtlhuber, David Hausheer and Anja Klein, "Energy-Efficient Data Dissemination in Ad Hoc Networks: Mechanism Design with Potential Game", International Symposium on Wireless Communication Systems (ISWCS), Brussels, 2015, pp. 616-620, doi: 10.1109/ISWCS.2015.7454421.

[2] Kevin Küchler, "Protokolldesign für den Aufbau energieeffizienter Broadcast-Bäume in kabellosen Ad-Hoc-Netzwerken", Master Thesis, SEEMOO, Technische Universität Darmstadt, Nov. 2020.