Dealing with *Bad Vibes* in *Open Airwaves*

Stefan Schmid (UPB)

Joint research with:
Andrea Richa (ASU)
Christian Scheideler (UPB)
Jin Zhang (ASU)

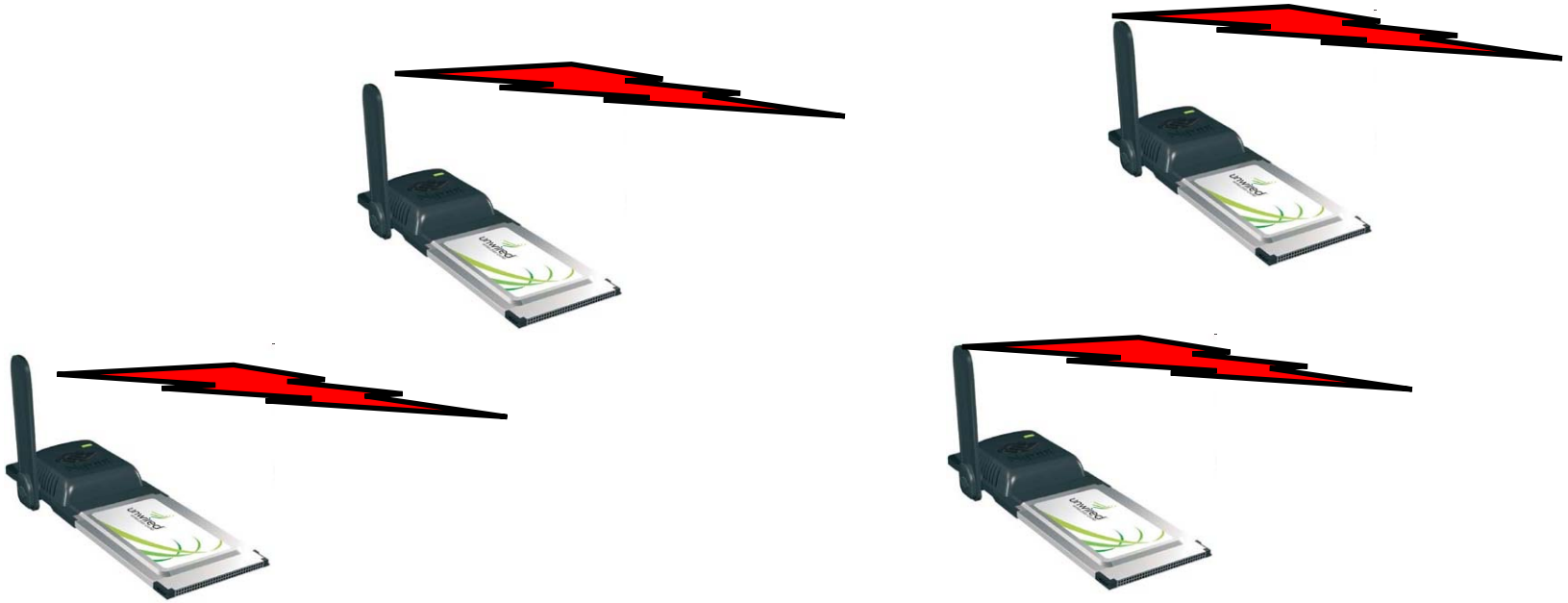
Open Airwaves

- A shared resource

- How to coordinate (efficient and/or fair) access?

Wireless Networks

- Medium Access Protocols needed



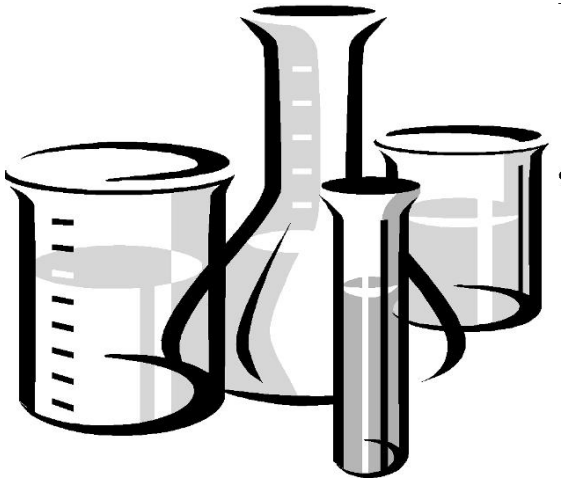
- Interference, collisions, ...
- ... and jammers.

Jammers („Bad Vibes“)

Often even without special hardward (cheap attack)!

This Talk: Recent Results on MacJam

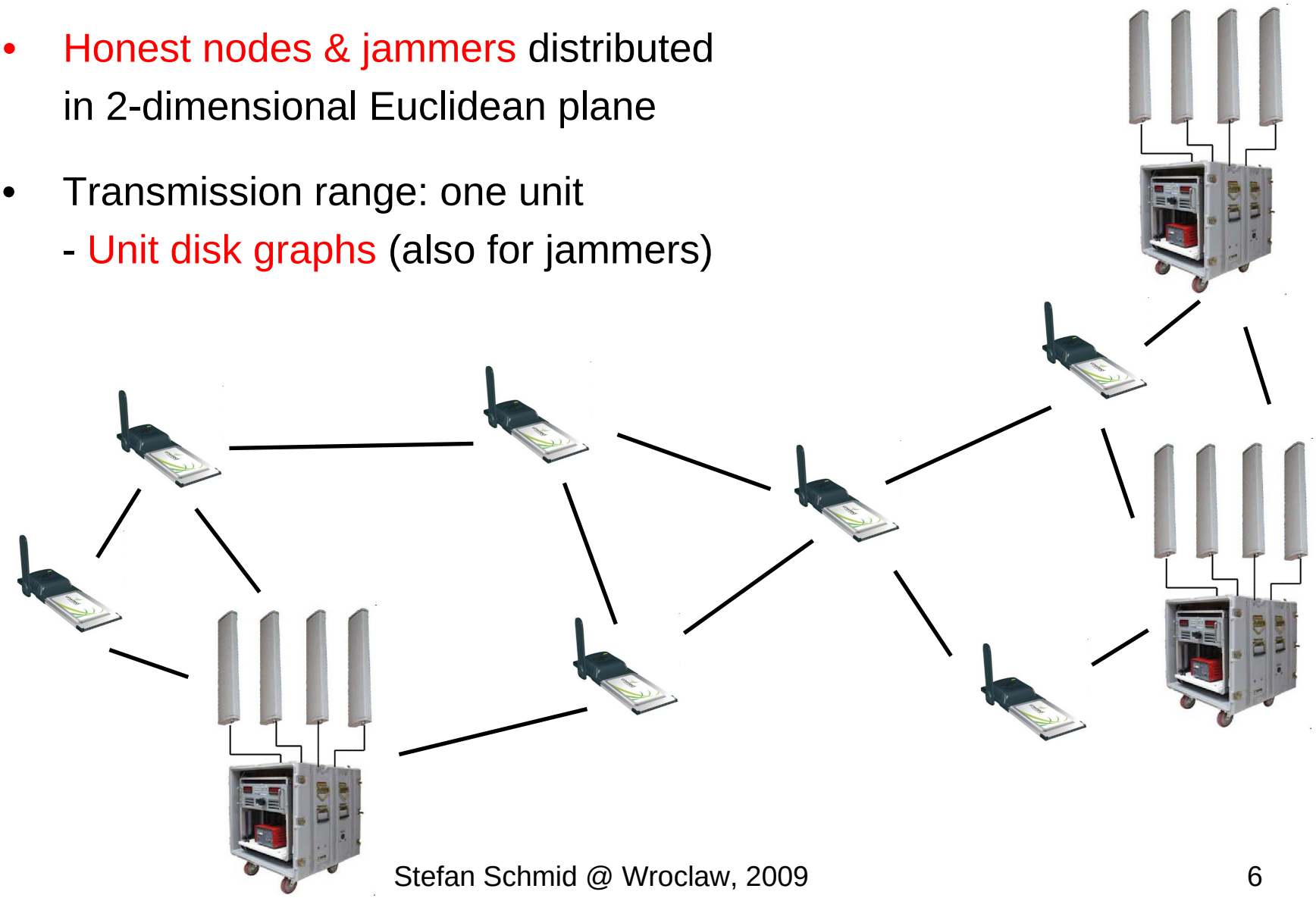
- Today‘s system not really jammer-proof (e.g., WLAN)
- MacJam = A robust MAC layer protocol
- Based on a randomized, distributed access strategy



- Attention:
  Preliminary/ongoing research!

Model: Multihop Wireless Network (1 Channel)

- Honest nodes & jammers distributed in 2-dimensional Euclidean plane
- Transmission range: one unit
  - Unit disk graphs (also for jammers)



Model: Multihop Wireless Network (1 Channel)

- Nodes:
  - Cannot send and listen concurrently (*one* antenna!)
  - Cannot distinguish between collision and jamming
  - Can recognize an idle channel
  - Backlogged: Always s.th. to send

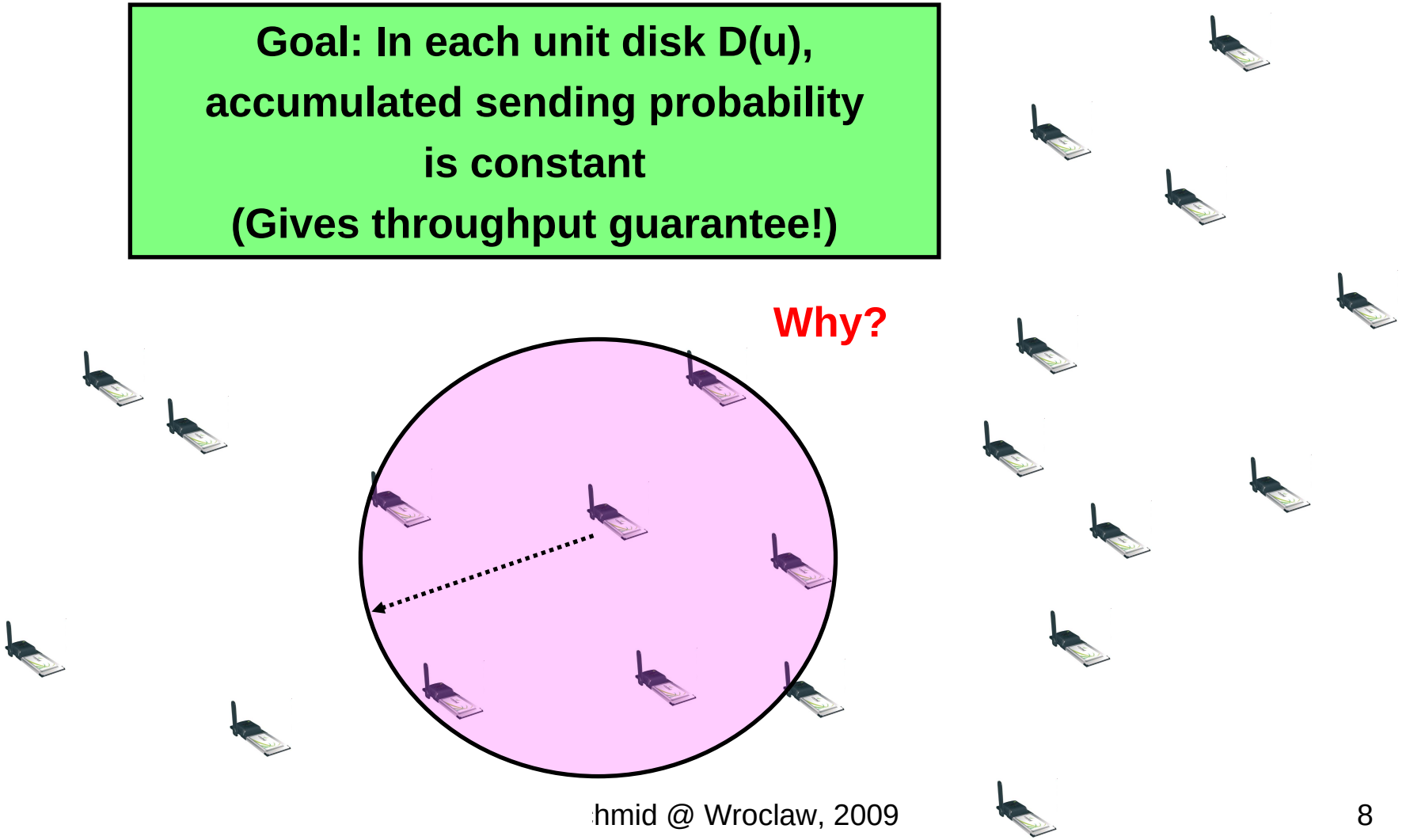
- Adversary:
  - Can jam a (1-ε) fraction of all time slots at all positions in the plane
  - Bursty (in time interval T, at most ε T)
  - Adaptive: Knows entire history (but not whether nodes will send in *this* round)

The MacJam Strategy (1)

- Each node v has sending probability p_v (adjusted dynamically)

Goal: In each unit disk D(u), accumulated sending probability is constant (Gives throughput guarantee!)

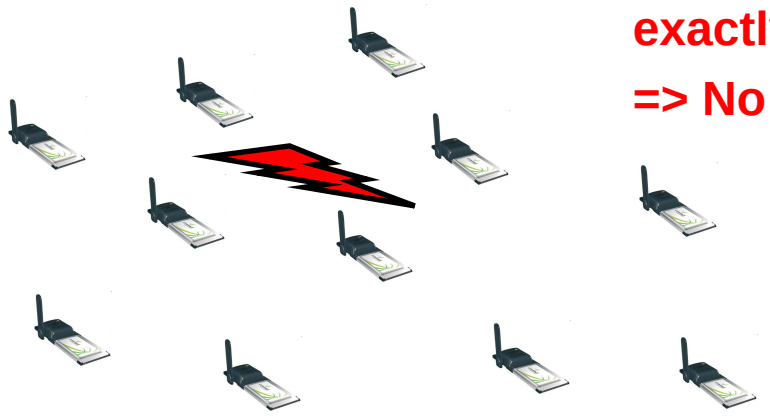
Why?



The MacJam Strategy (1)

- Each node v has sending probability p_v (adjusted dynamically)

Goal: In each unit disk D(u), accumulated sending probability is constant (Gives throughput guarantee!)



Clique: we expect exactly one node sends => No collision & good throughput

The MacJam Strategy (2)

- Example with clique: How to achieve a constant probability?
- Let $\mathbf{q_0}$ be prob. that channel is idle,
  and $\mathbf{q}_1$ be prob. that exactly one successful transmission
  hat(p)=const the max node probability, p the total sending prob.

  It holds: $q_0 = \prod_v (1-p_v)$, $q_1 = \sum_v p_v \prod_{q \neq v} (1-p_w)$

  It can be shown: $q_0 \cdot p \leq q_1 \leq q_0 / (1- \text{hat}(p)) * p$

So what?

Thus: If # idle slots = # successful slots => $p=\sum_v p_v \approx 1$

Algorithm can be independent of collisions/jammed rounds!
(If there are enough idle and successful slots..)

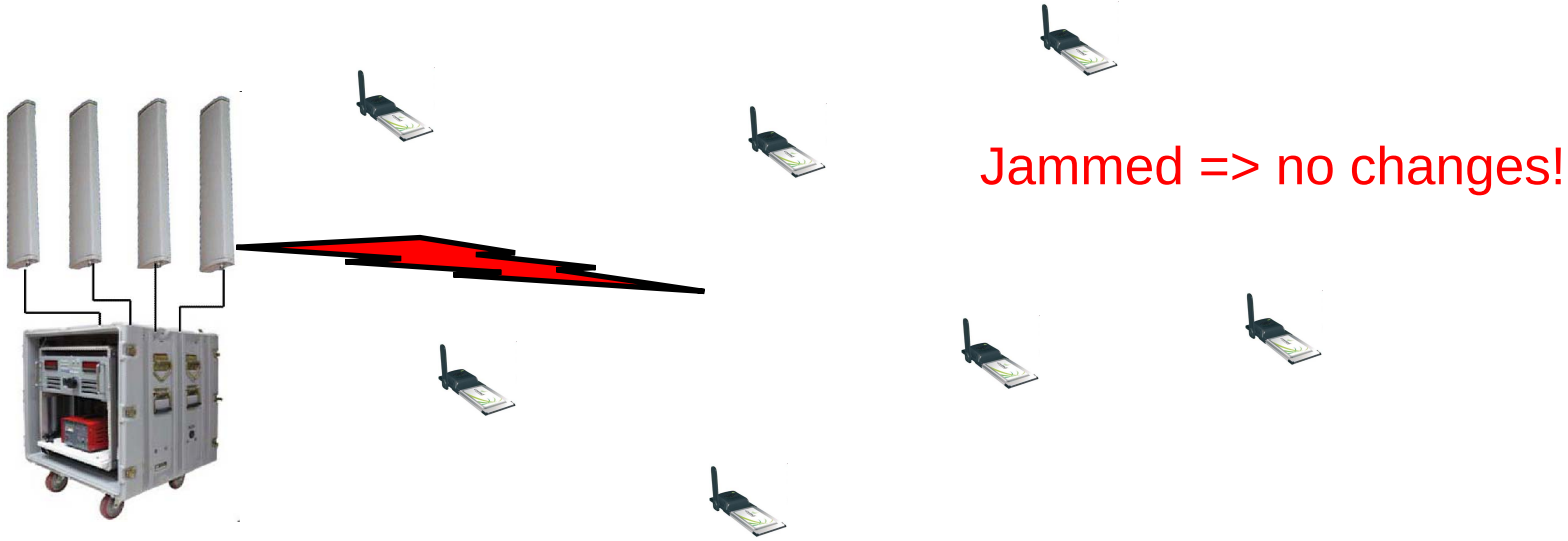
The MacJam Strategy (2)

- (Too) simple MAC protocol (for some $\gamma>0$):

If (idle): $p_v := (1+\gamma)\ p_v$

If (success): $p_v := 1/(1+\gamma)\ p_v$

- Example with clique:



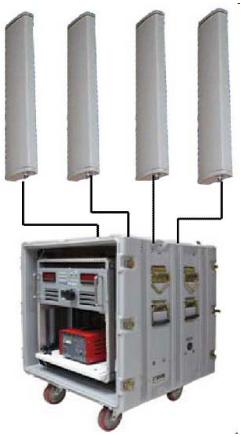
The MacJam Strategy (2)

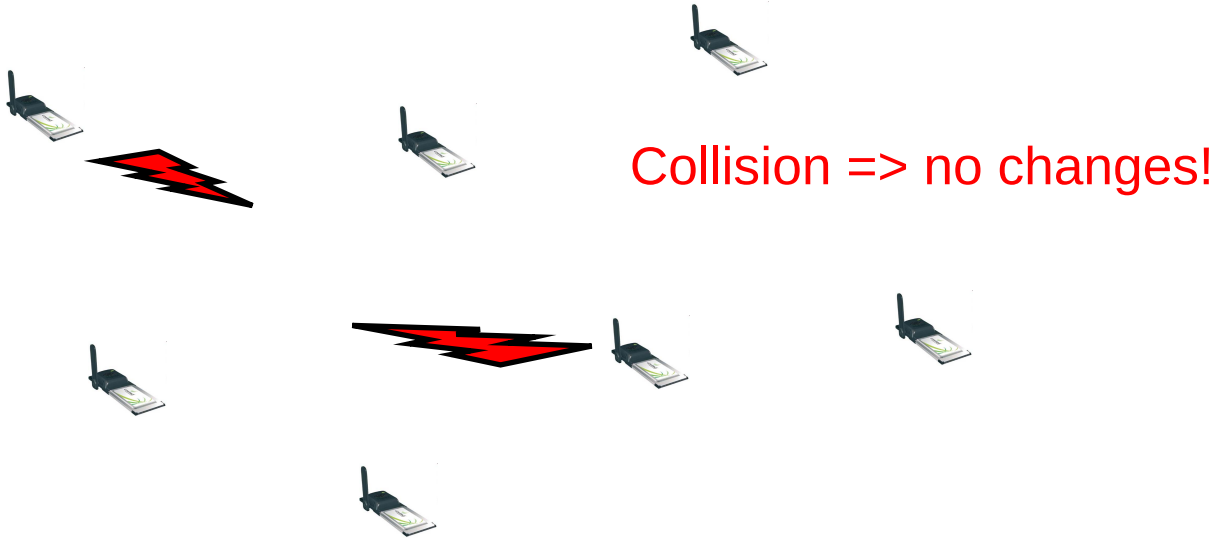
- (Too) simple MAC protocol (for some γ>0):

If (idle): $p_v := (1+\gamma)\ p_v$

If (success): $p_v := 1/(1+\gamma)\ p_v$

- Example with clique:





Collision => no changes!

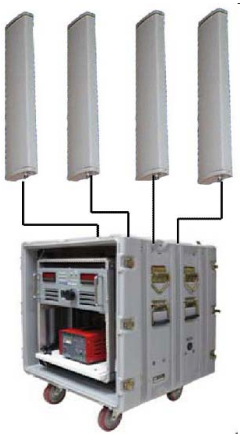
The MacJam Strategy (2)

- (Too) simple MAC protocol (for some γ>0):

If (idle): $p_v := (1+\gamma)\ p_v$

If (success): $p_v := 1/(1+\gamma)\ p_v$

- Example with clique:



Idle => increase!

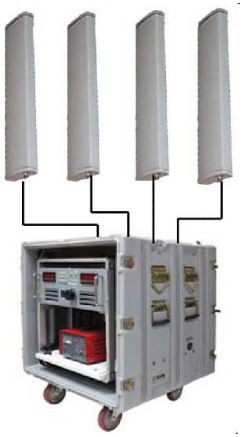
The MacJam Strategy (2)

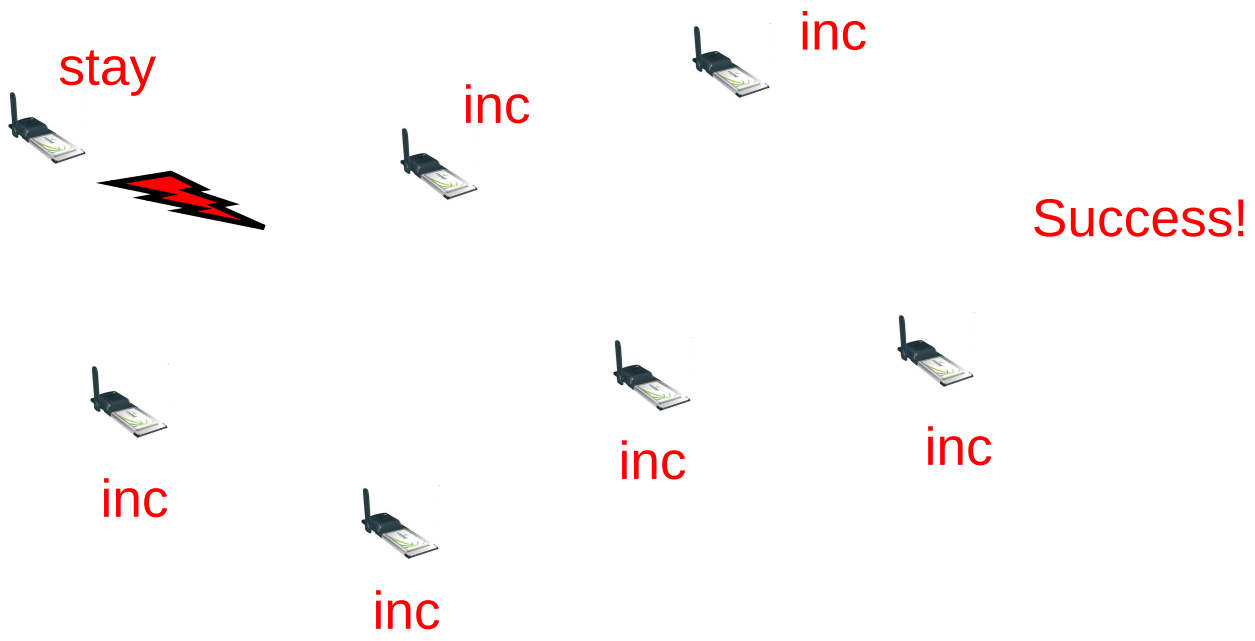
- (Too) simple MAC protocol (for some γ>0):

If (idle): $p_v := (1+\gamma)\ p_v$

If (success): $p_v := 1/(1+\gamma)\ p_v$

- Example with clique:





The MacJam Strategy (3)

- Problem?

If (idle): $p_v := (1+\gamma)\, p_v$

If (success): $p_v := 1/(1+\gamma)\, p_v$

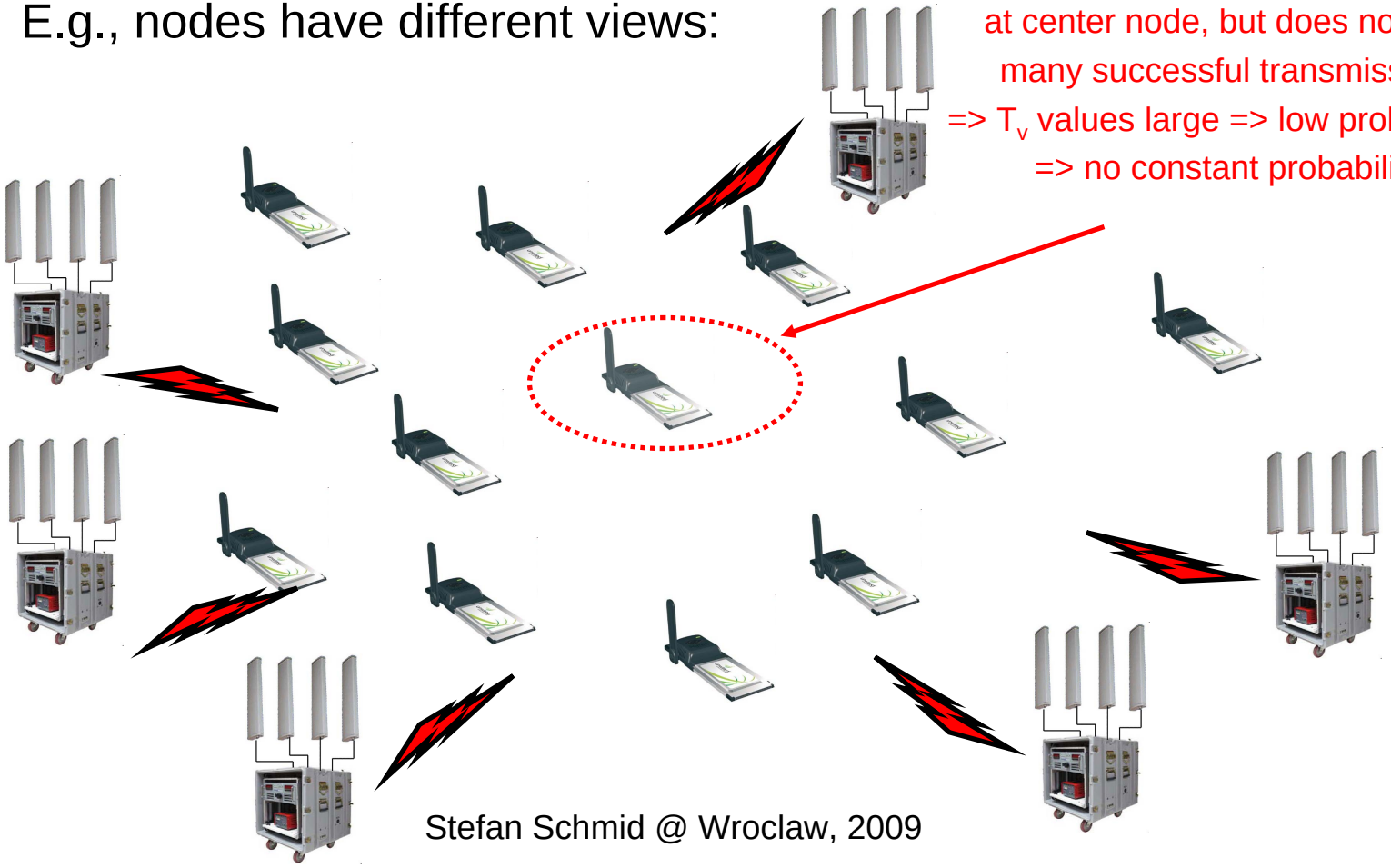
- Problem: if p_v initially very high, there are hardly any idling or successful slots to observe!

- Therefore: introduce a threshold T_v
  - if no successful transmission within T_v, decrease p_v

Singlehop vs Multihop

- Compared to earlier work, multi-hop setting is more complex (algo & analysis)

- E.g., nodes have different views:



Nodes around center are jammed
=> high sending probability
at center node, but does not hear
many successful transmissions
=> T_v values large => low prob as well
=> no constant probability!

The MacJam Strategy (4)

- The MacJam protocol (for UDGs):

T_v=1, c_v=1, $p_v = p_{max}$;
In each round:
  decide to send with prob p_v;
  if decide not to send:
    if sense *idle channel*: $p_v = (1+\gamma)\, p_v$; T_v--;
    if *succ* reception: $p_v = 1/(1+\gamma)\, p_v$; T_v--;
  c_v++;
  if ($c_v > T_v$)
    c_v=1;
    if no *idle* or *succ* in last T_v steps:
      $p_v = 1/(1+\gamma)\, p_v$; $T_v = T_v + 2$;

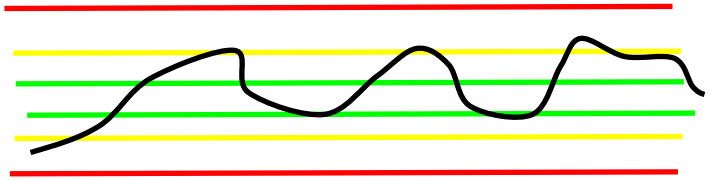
New: idle is okay, too!

New: +2 (not +1)

Analysis (1)

- Some „ideas“ only
- Protocol is interplay of many dependent randomized local algorithms
- Cumulative probability thresholds:

ρ_{green}, ρ_{yellow}, ρ_{red}

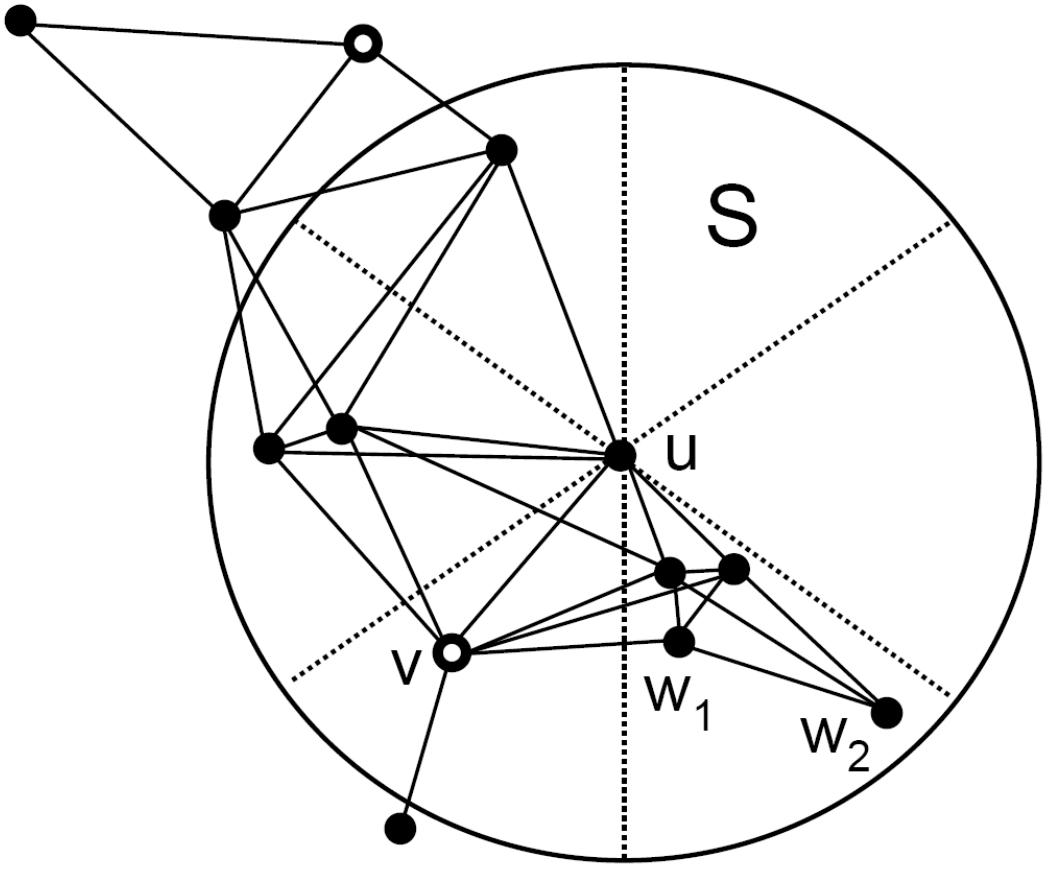


Show that beyond „good accumulated probabilities“, there is a high drift towards „better values“

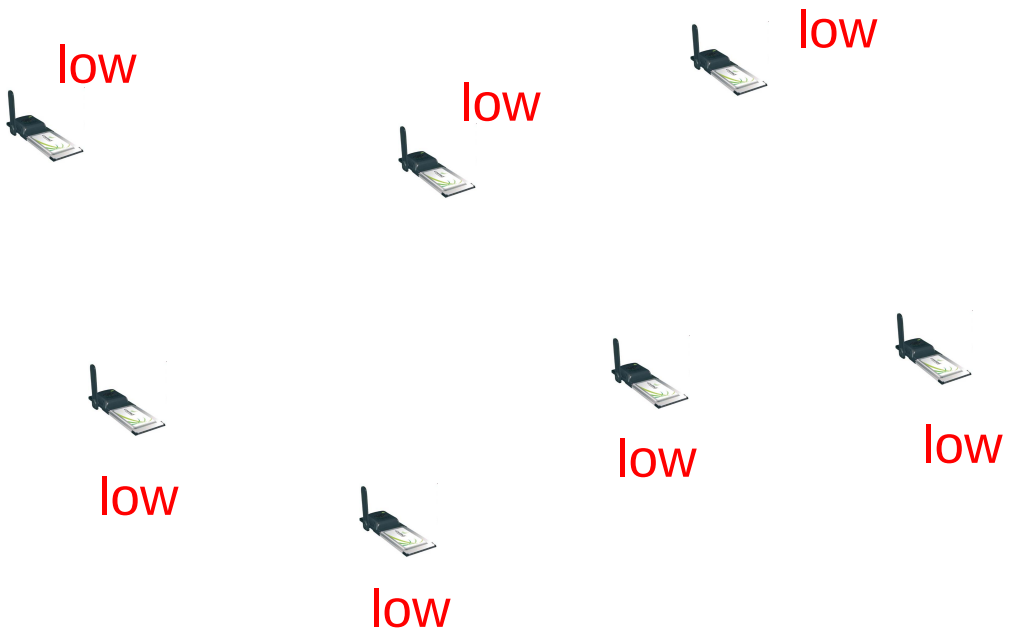
- Techniques: Martingale theory, stochastic dominance, etc.

Analysis (2)

- Idea: Consider sectors of completely connected networks (PODC)



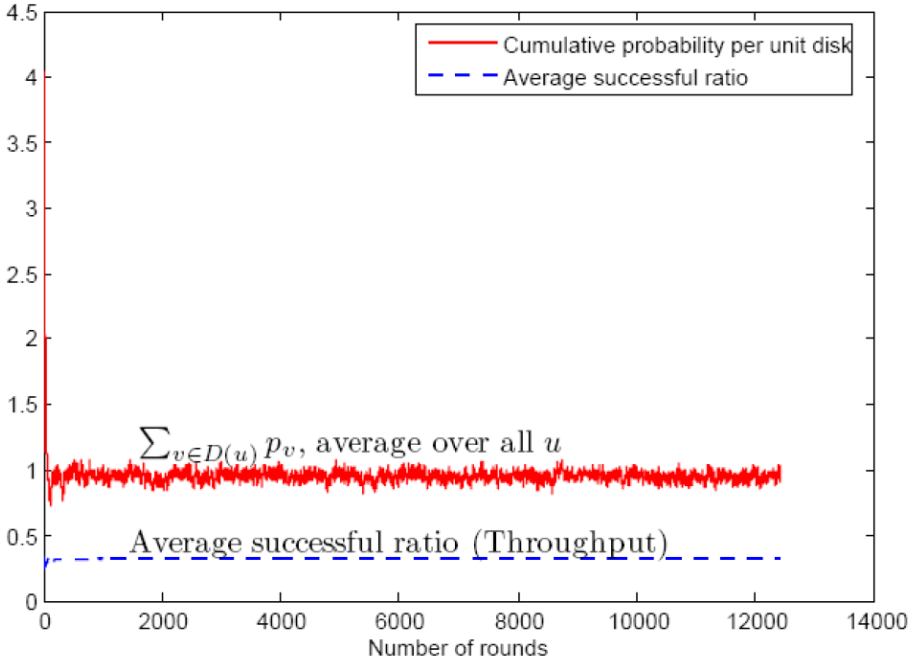
Analysis



Many idle slots => fast probability increase!

Simulations

- 500 nodes uniformly at random in 4x4 plane, $\varepsilon \approx 0.5$
- Converges fast to good cumulative probabilities
- Around 30% of unjammed slots are successful transmissions
- T_v values around 2 or 3



Ideas for Extensions/Applications

- How to make it fair?
  - Problem: When a node is successful, other nodes will reduce p_v => node may be even more successful in future
  - Solution: Nodes remember number of nodes seen so far, and maintain a counter for successful transmissions.
  - Adapt their probabilities in a more equal manner (all around 1/n in clique)!

- Leader Election
  - Contention resolution with MacJam
  - Leaders increase sending probability faster (to constant!)
  - Dedicated leader slots determined online
  - When leader offline: new one is selected

Conclusion & Future Work

- Jammers exciting research challenge
  - May improve robustness and performance in existing networks

- Many open questions
  - Provable MAC performance
  - Multihop networks
  - Fairness
  - Energy Efficiency
  - Applications

- But we are working on it... ☺

Thank you for your attention!

More infos on:

http://www.cs.uni-paderborn.de/fachgebiete/fg-ti/personen/schmiste.html