

# Byzantine Resilient Synchronization for Content and Presence Updates in MANETs

**Bogdan Carbunar**

**M. Pearce, S. Mohapatra, L. Rittle, V. Vasudevan**

**Applications Research and Technology Center  
Motorola**



# MeCast

Bob



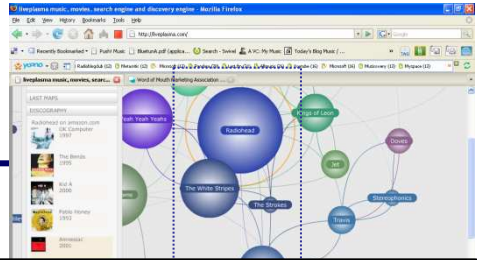
*Karma Police*  
Radiohead  
*Sulk*  
Radiohead



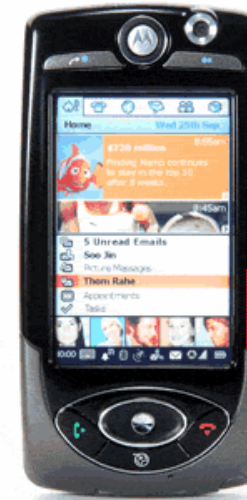
*Pump it up*  
Elvis Costello

Identify an interesting source

Music Surveillance



Alice



*Amsterdam*  
Coldplay  
*Crash*  
Dave Matthews

caster

castee



*All we have is now*  
Flaming Lips



*Jump*  
Madonna

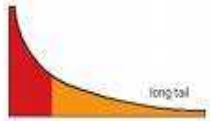


# MeCast

Bob



*Karma Police*  
Radiohead  
*Sulk*  
Radiohead  
*Finally we are no one*  
Mum (Iceland)



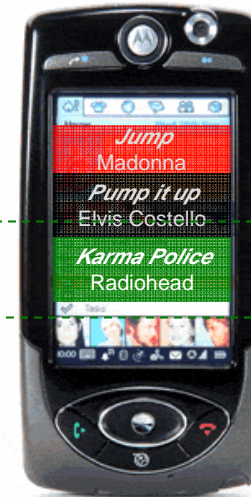
caster

castee



Tune-in to it for a bit

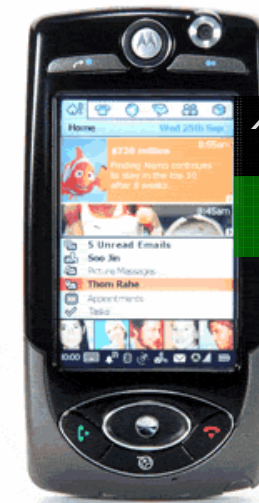
Alice



# Content & Presence Multicast Protocol

## CPMP: messaging format

- Periodic transmission over IP multicast
- Disseminate content and presence updates
- [CPMP, Id, T<sub>x</sub>, metadata]



Bob

Amsterdam  
ColdPlay  
Sulk  
Radiohead

[CPMP, Bob, 20'',  
Amsterdam.ColdPlay:4'20'']

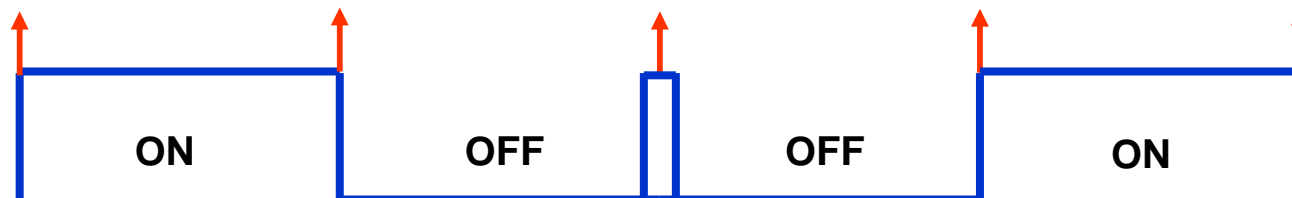
[CPMP, Bob, 20'',  
Amsterdam.ColdPlay:4'0'']

[CPMP, Bob, 20'',  
Sulk.Radiohead:3'28'']



# Challenges

- **Routing of updates is expensive (requires cooperation, incentives)**
  - Do not route updates
  - Only collect updates from devices within transmission range
- **Always-on Wi-Fi is power consuming**
  - Wi-Fi duty cycle: e.g., each minute the Wi-Fi is on for 20s, off for 40s
  - Send updates periodically
  - While off, the Wi-Fi wakes up periodically (e.g., 20s) for 1-2s to transmit/receive updates



**Neighbor updates will be lost when Wi-Fi is off !**



# Solution Properties

## P1. Send updates in sync with neighbors

- Not required to sync duty cycles

## P2. Distributed

- No central coordination point

## P3. Efficient

- Do not require additional communication or large updates

## P4. Timely

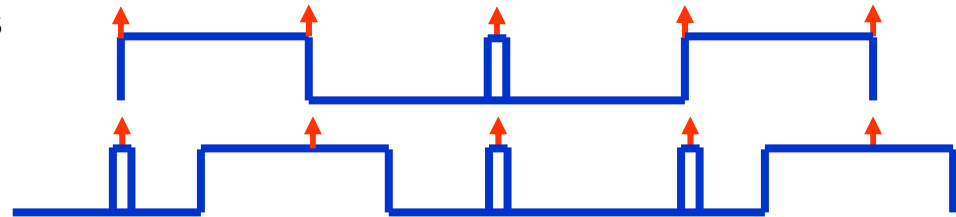
- Devices are co-located for short intervals

## P5. Resilient to Byzantine Attacks

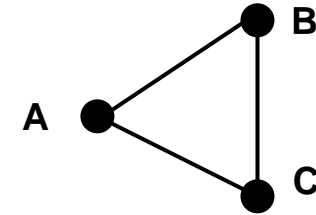
- Attacker may corrupt devices, change the contents of CPMP packets, spoof device identifiers

## P6. Simple

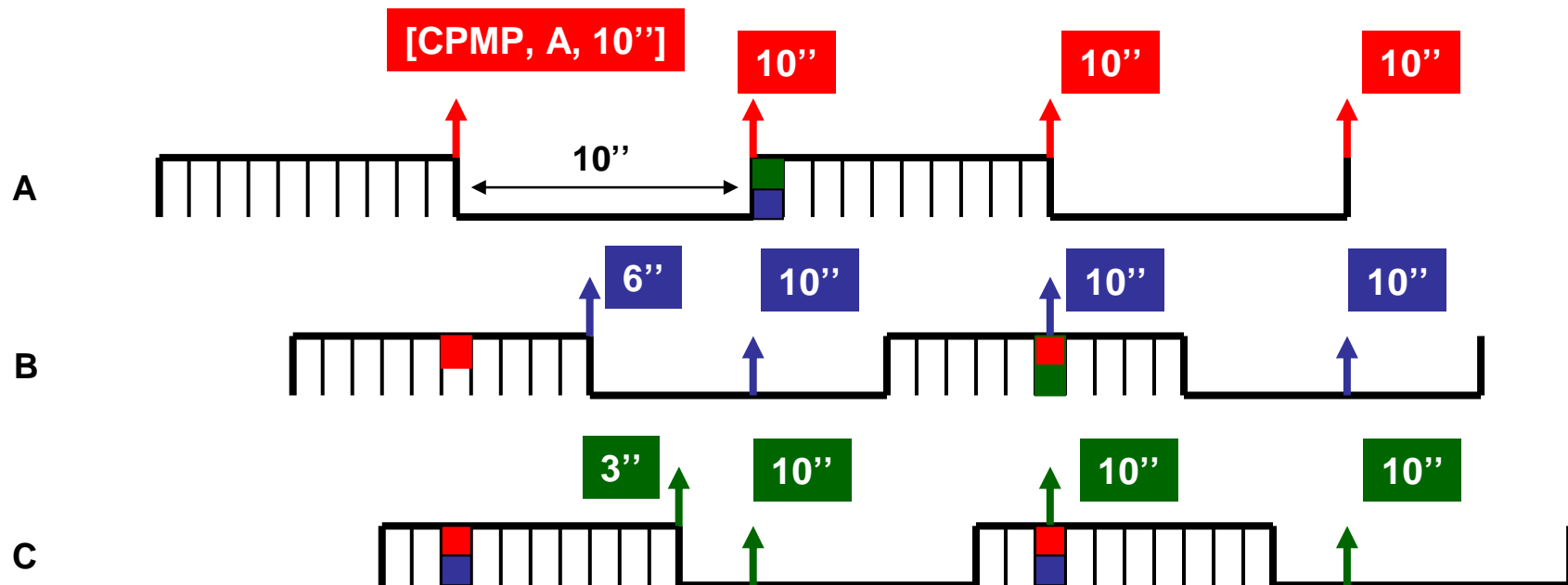
- Do not use expensive crypto (PKI, certificates)



# Future Peak Detection - FPD

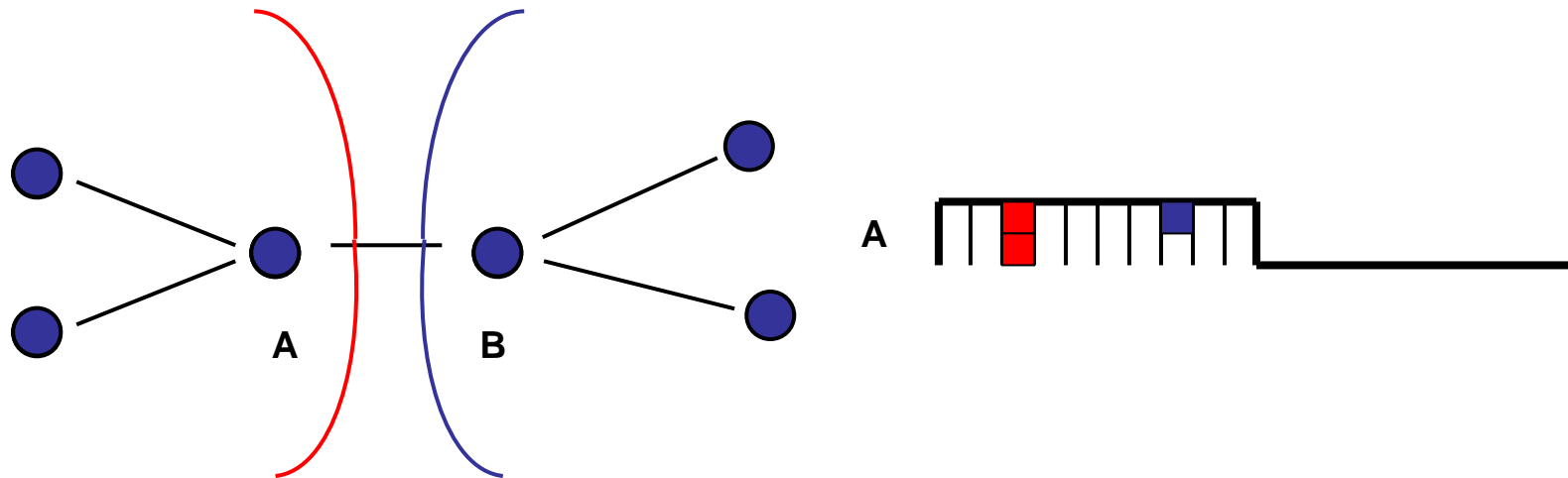


- Divide each active interval into slots
- Place each neighbor update = [CPMP, Id,  $T_x$ , metadata] in one slot
  - $\text{Slot}(\text{update}) = (\text{current\_time} + T_x) \bmod N\_slots$
- Synchronize transmission with slot containing most packets



# Randomized Future Peak Detection - FPDR

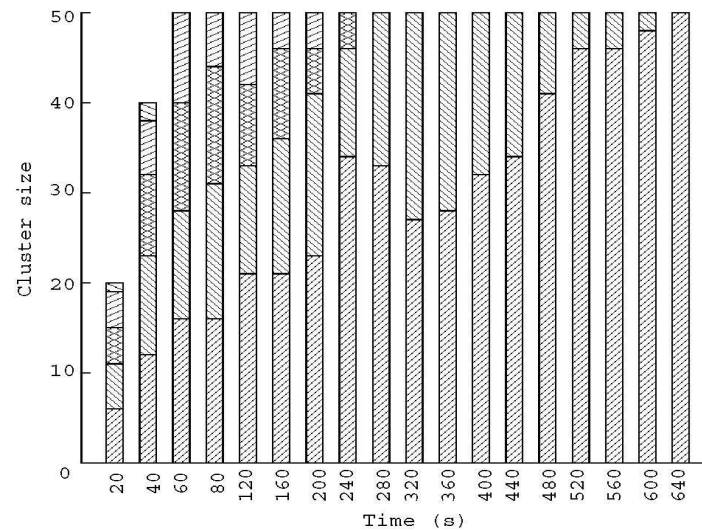
- **Problem: greedy choice of synchronization slot leads to network partitions**



- **FPDR: weighted choice of slots -- A chooses slot 3 with probability  $2/3$  and slot 8 with probability  $1/3$**

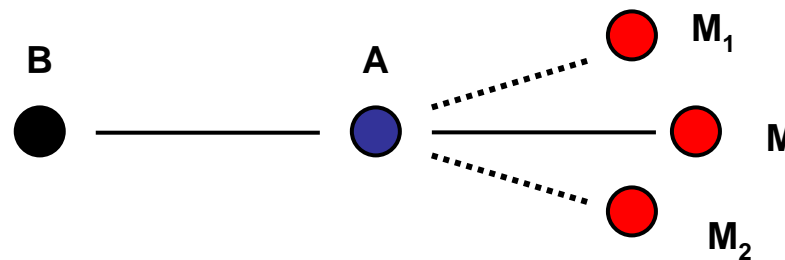
# Performance

- Simulation: 50 nodes deployed in 150 x 150 m<sup>2</sup> area
- Motorola A910 phone – 30 m transmission range
- Duty cycle: active interval - 5s followed by 3 sleep intervals - 5s each
- Nodes join in quick succession – 1 per second
- Synchronization takes 10'
- Only two clusters left after 4'



# Attacks against FPD/R

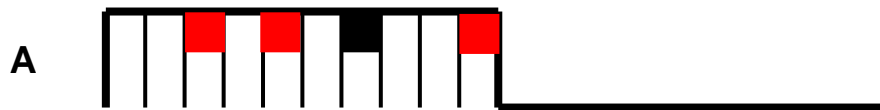
- FPD/R rely on minimal information from updates - arrival time &  $T_x$
- Attacker can spoof device ids and generate multiple updates



- **Tower attack:** all updates fall in the same slot



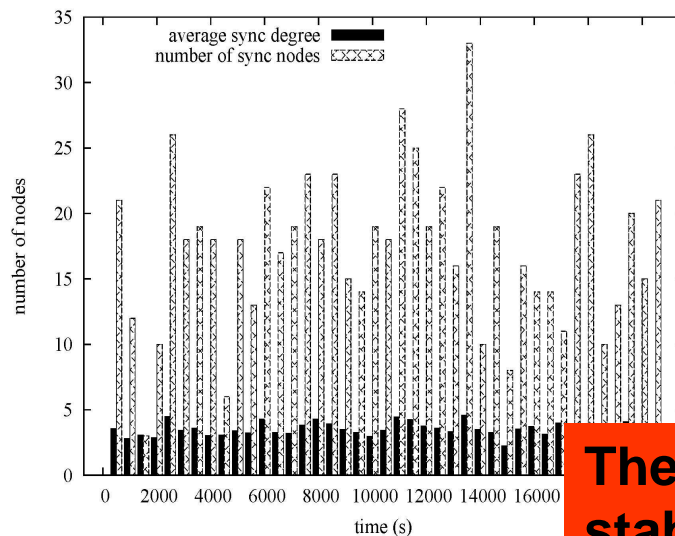
- **Dispersion attack:** updates are distributed uniformly over slots



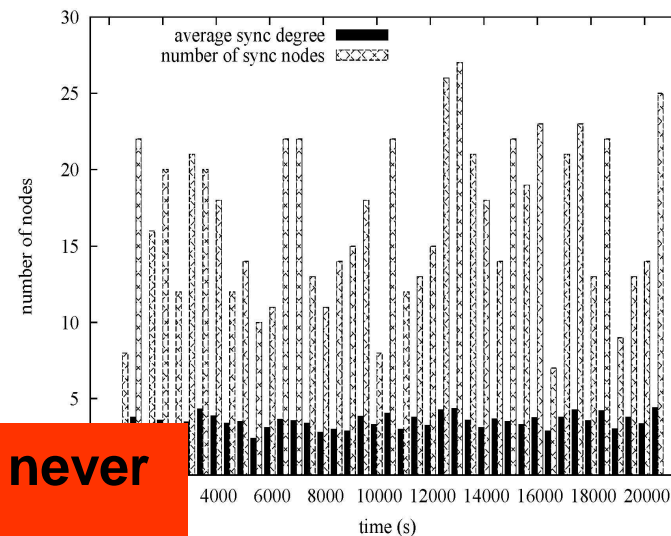
**Different neighbors of M will sync with different slots !**

# Effects of Attacks on FPDR

- 50 nodes deployed in 150 x 150 m<sup>2</sup> area
- One malicious node deployed at the center (first to join)
  - Sends 10 packets during each 5s interval
- **Average sync degree**: number of neighbors a node is synced with
- **Fully synced nodes**: number of nodes synced with all neighbors



Tower attack

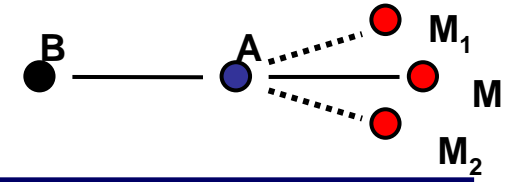


Dispersion attack

The network never stabilizes !

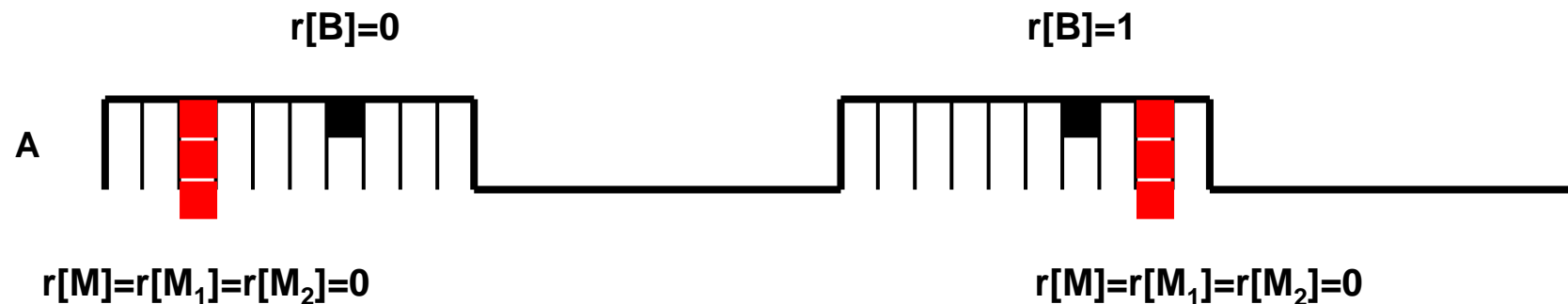


# Rating Based Algorithm - RBA



- Each node builds a reputation for each device id seen during last active interval
- Consistency in choosing slots is rewarded, switching is severely punished

- Newly seen ids receive reputation 0 ( $r=0$ )
- When update from neighbor falls in the same slot,  $r++$
- When update placed in a different slot,  $r=0$

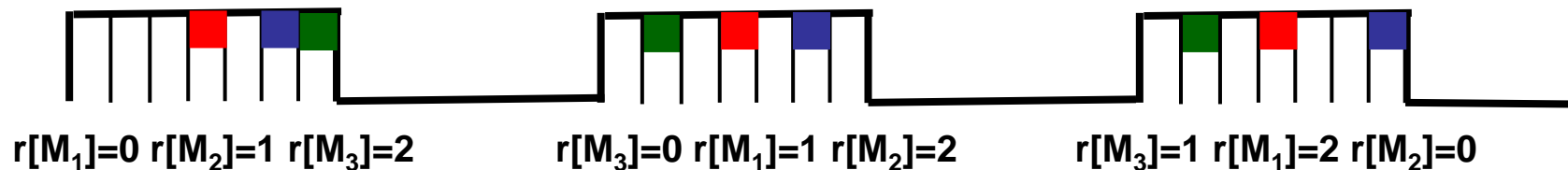


- RBA: sync with slot holding update from neighbor with highest reputation



# Circular Cascade Attack

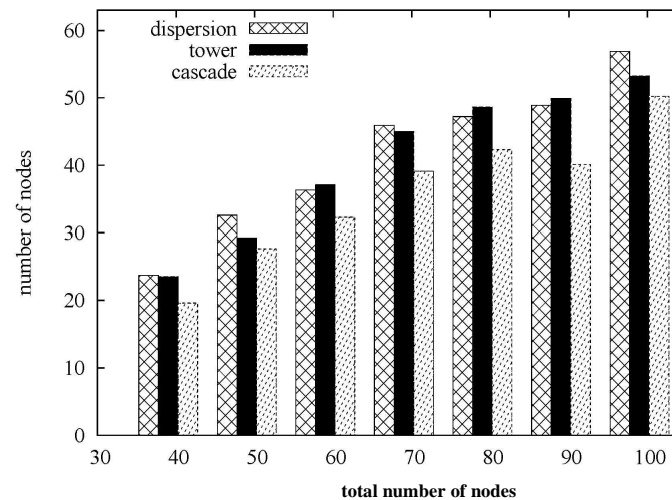
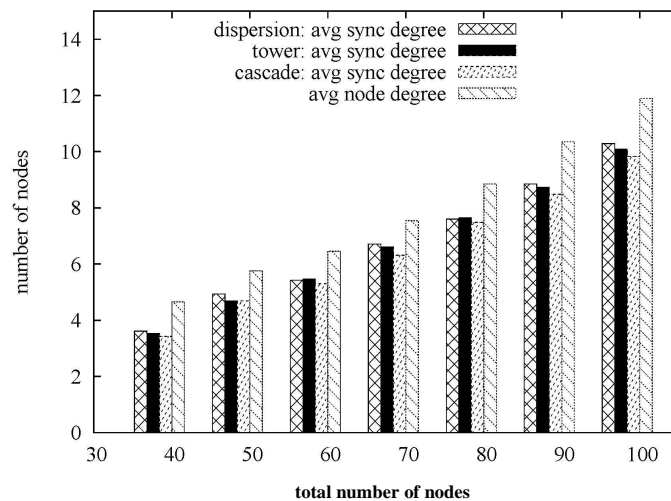
- M chooses  $n$  ( $=3$ ) device ids  $M_1, M_2, M_3$ 
  - $r[M_1]=0, r[M_2]=1, r[M_3]=2$
  - When M's neighbors have synced with  $M_3$ , change  $M_3$ 's slot –  $r[M_3]=0$
  - Everyone re-syncs with  $M_2$  ( $r[M_2]=2$ )
  - repeat process



- Outcome: M's neighbors will follow  $M_3$  then  $M_2$ , then  $M_1$  then again  $M_3$  ...
- RBA:
  - M's neighbors may have more stable neighbors than  $M_1, M_2, M_3$
  - the attack can only succeed on M's neighbors

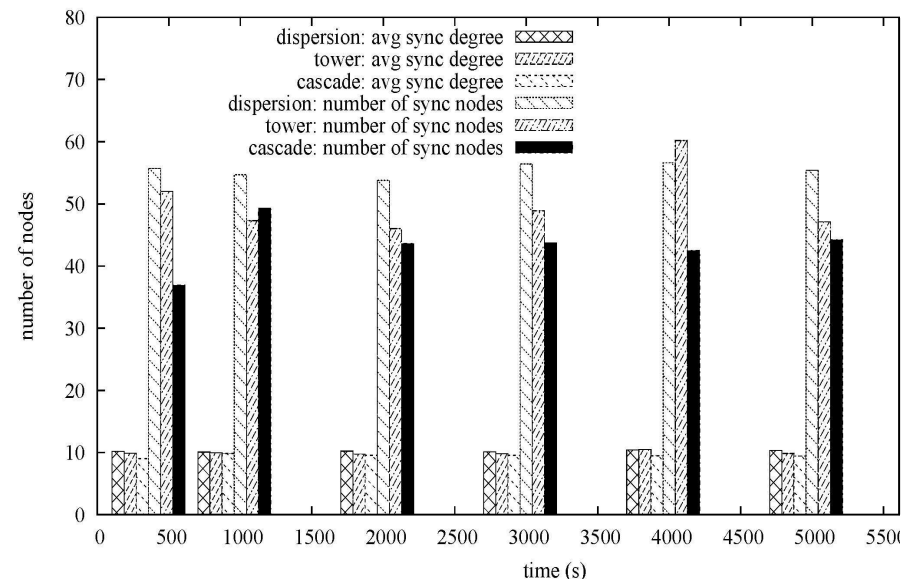
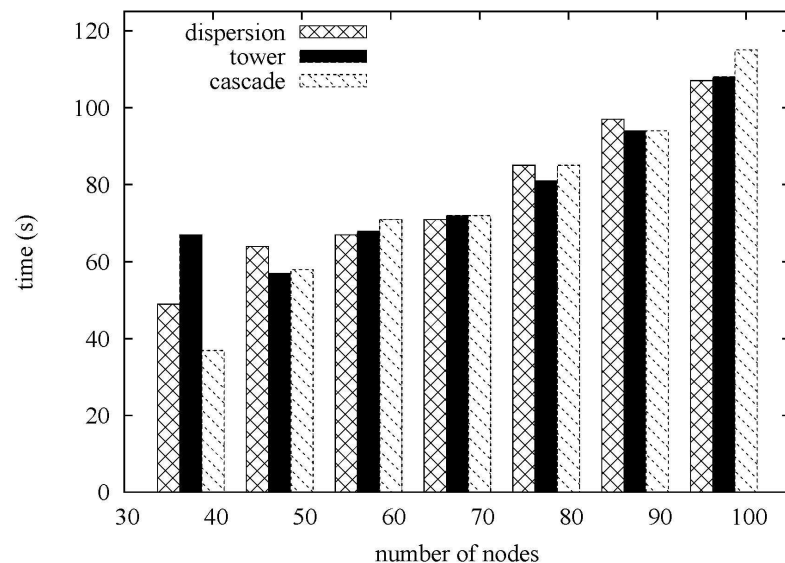
# RBA's resilience to attacks

- **First experiment: networks between 40 to 100 nodes**
- **Cascade attack: 10 spoofed ids, ratings 0,1,...,9**
- **Strategy: run FPDR for 1000s to build reputations then switch to RBA**
- **RBA's greedy choice partitions the network**
- **Average sync degree: nodes are synced with almost all neighbors**
- **Number of sync nodes: tower & dispersion 55-65%, cascade 44-55%**
- **Average update loss rate: 15% for a 100 node network**



# Resilience to attacks (cont'd)

- Time to stabilize - even for 100 nodes is less than 120s
- Second experiment: effects of history size on RBA performance
- 100 node network, change time to run FPDR
- Running FPDR for 400s is sufficient



# Conclusion

- **Power savings of devices exchanging periodic information can be achieved using synchronization of transmission times**
- **Synchronization needs to be resilient to Byzantine behavior of nodes**
  - **Simple Design:** more difficult to attack
    - **Minimum transmission overhead – piggyback synchronization information on existing updates**
  - **Do not trust information contained in updates**
    - **do not propagate information received in updates**
    - **use implicit information: packet arrival times and stability of transmissions**
- **FPDR syncs the whole network but vulnerable to attacks**
- **RBA may partition the network but stabilizes quickly under attack**



# Questions ?



# Performance

- Motorola A910 phone
- Duty cycle: active interval - 5s followed by 3 sleep intervals - 5s each
- **30% lifetime extension over Wi-Fi always on**

