Design and Evaluation of a new MAC Protocol for Long-Distance 802.11 Mesh Networks

by

Bhaskaran Raman & Kameswari Chebrolu

ACM Mobicom 2005

Reviewed by

Anupama Guha Thakurta

CS525M - Mobile and Ubiquitous Computing Seminar, Spring 2006

OUTLINE

- Introduction
- Background
- Protocol Design and Implementation
- Topology Construction
- Evaluation
- Discussion and Conclusions
- Comments

INTRODUCTION

- **Motivations for new protocol:**
  - low cost internet access to rural areas
  - achieve performance improvement over 802.11 CSMA/CA in long distance mesh networks
- **802.11 CSMA/CA MAC was designed to resolve contentions in indoor environments**
- **Use of wire-line, cellular or 802.16 currently prohibitive because of costs**

INTRODUCTION (Cont.):
Issues Addressed

- **Find an alternative to 802.11 CSMA/CA MAC protocol that allows simultaneous synchronous transmission / reception of multiple links at single node**
- **Propose a new MAC protocol: 2P**
  - Cost advantages with off-the-shelf 802.11 hardware
  - Show dependence of 2P on network topology
  - Show that more UDP throughput than CSMA/CA is achievable (achieved 3-4 times)
  - Show that more TCP throughput than CSMA/CA is achievable (achieved 20 times)

INTRODUCTION (Cont.):

Mesh NW Characteristics

- Multiple radios per node (one radio per link)
- High-gain directional antennae
- Long distance point-to-point links of several kilometers
    - **Landline node**

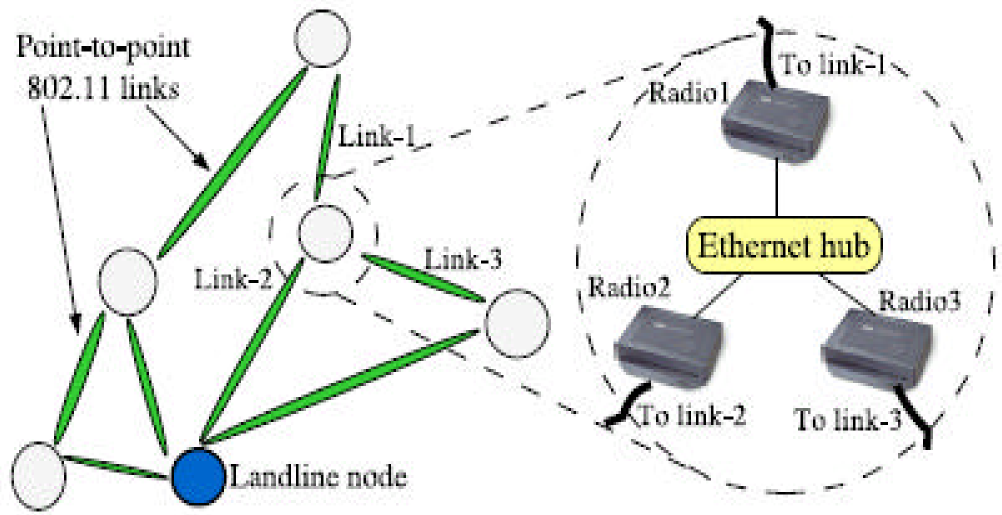


Figure 1. 802.11 mesh: nodes have one radio per link

OUTLINE

- Introduction
- Background
- Protocol Design and Implementation
- Topology Construction
- Evaluation
- Discussion and Conclusions
- Comments

BACKGROUND

SynOp: Simultaneous Synchronous Operation (SynRx / SynTx)

- **Syn-Rx:** R1 and R2 receive simultaneously; Feasible
- **Syn-Tx:** T1 and T2 transmit simultaneously; Feasible
- **Mix-Rx-Tx:** R1 receives and T2 transmits; Not feasible

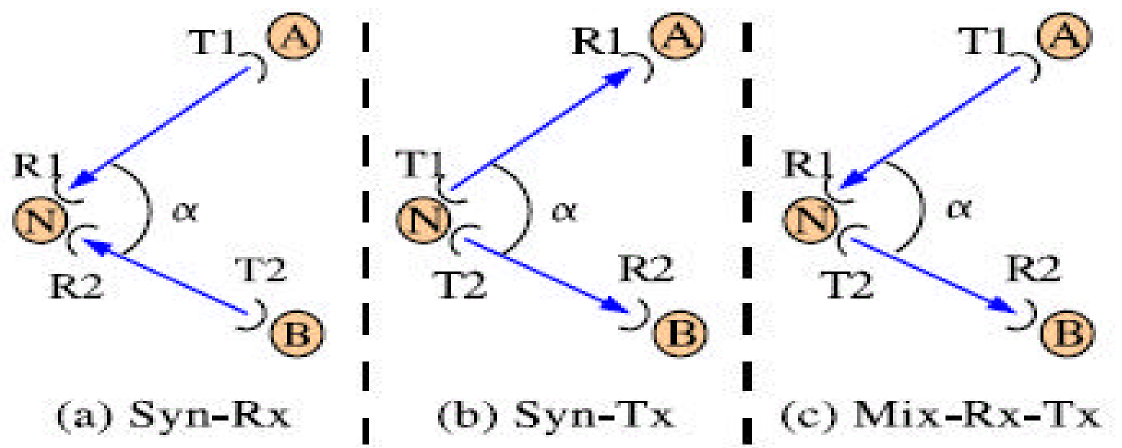


Figure 2. SynRx, SynTx, and Mix-Rx-Tx

BACKGROUND (Cont.):

SynOp: Simultaneous Synchronous Operation (SynRx / SynTx)

- **In 802.11 Mix-Rx-Tx is not feasible because of:**
  - ✓ physical proximity and side lobes of directional antennae
- **In 802.11 SynOp is feasible but not allowed because:**
  - ✓ SynRx: IFS based immediate ACK mechanism
  - ✓ SynTx: Carrier sense mechanism of interfaces give rise to backoffs

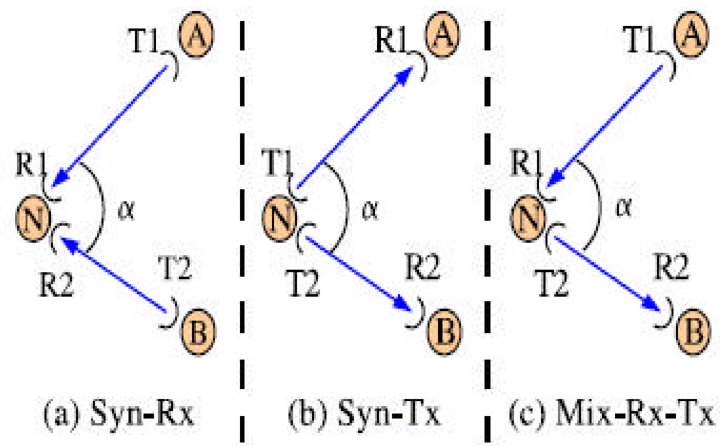


Figure 2. SynRx, SynTx, and Mix-Rx-Tx

OUTLINE

- Introduction
- Background
- Protocol Design and Implementation
- Topology Construction
- Evaluation
- Discussion and Conclusions
- Comments

2P PROTOCOL DESIGN & IMPLEMENTATION

- SynOp is possible by disabling ACK and Carrier sense mechanisms
- Simple Concept: each node switches between SynRx & SynTx
- When a node is in SynRx its neighbors are in SynTx phase and vice the versa
- Bipartite Topology

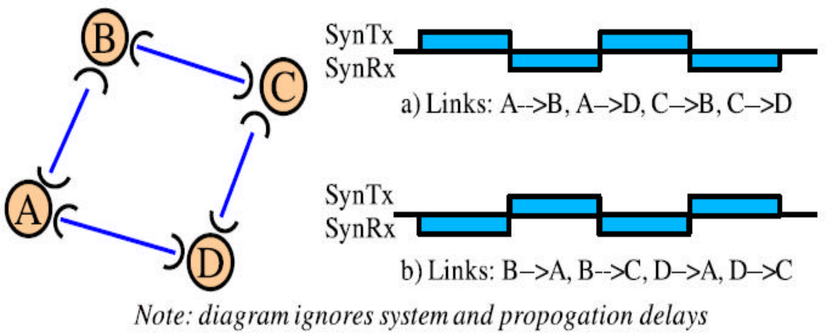


Figure 4. 2P Illustration

2P PROTOCOL DESIGN & IMPLEMENTATION (Cont.):

- **Solutions for SynRx in existing hardware:**

Disable immediate ACKs' by:

- Independent Basic Service Set mode for interface operations, with separate SSID
- Convert IP unicast pkts. to MAC broadcast pkts. at the driver level
- Send ACKs' in the LLC implemented by the driver, by piggybacking them on data packets

2P PROTOCOL DESIGN & IMPLEMENTATION (Cont.):

- **Solutions for SynTx in existing hardware:**

Disable carrier-sense backoffs by:

- utilizing the two antennae connector feature provided by Intersil Prism chipset

How it works:

- ✓ Select receiving antenna at driver level by *antsel_rx* command
- ✓ Connect external antenna to, say LEFT connector of radio card
- ✓ During transmission, the receiving antenna connector which is not connected to any external antenna is set to RIGHT
- ✓ This forces carrier-sense to happen on the RIGHT connector which sees only negligible noise
- ✓ Switch the receiving antenna to LEFT connector before switching from SynTx to SynRx

OVERHEAD?

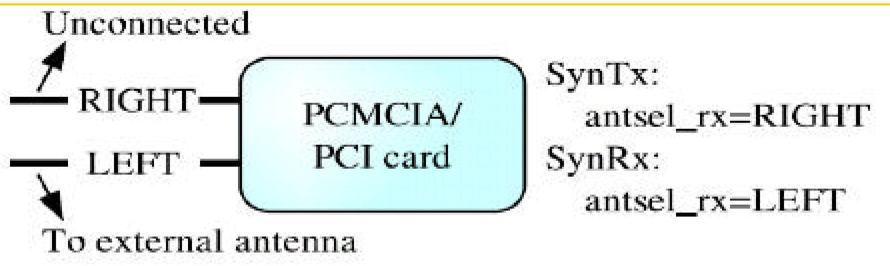
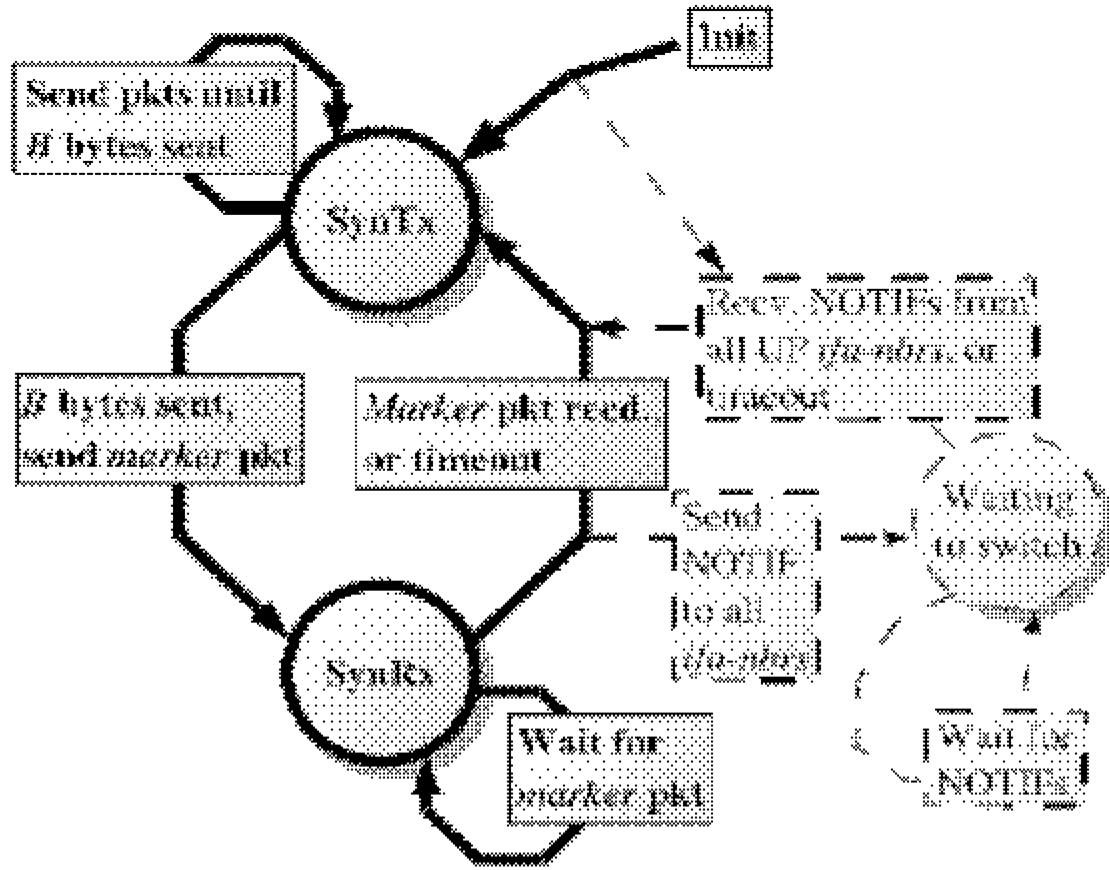


Figure 5. Using *antsel_rx* to avoid carrier-sensing

2P PROTOCOL DESIGN & IMPLEMENTATION (Cont.): Loose Synchrony

- An interface sends B bytes in SynTx, then sends a m*arker* packet as a "token"
- Enter the SynRx phase
- Switch to SynTx upon receiving a marker packet or upon timeout

OVERHEAD?



2P PROTOCOL DESIGN & IMPLEMENTATION (Cont.): Problems in Loose Synchrony

- Temporary loss of synchrony (marker loss)
- Link intialisation (link recovery after failure)

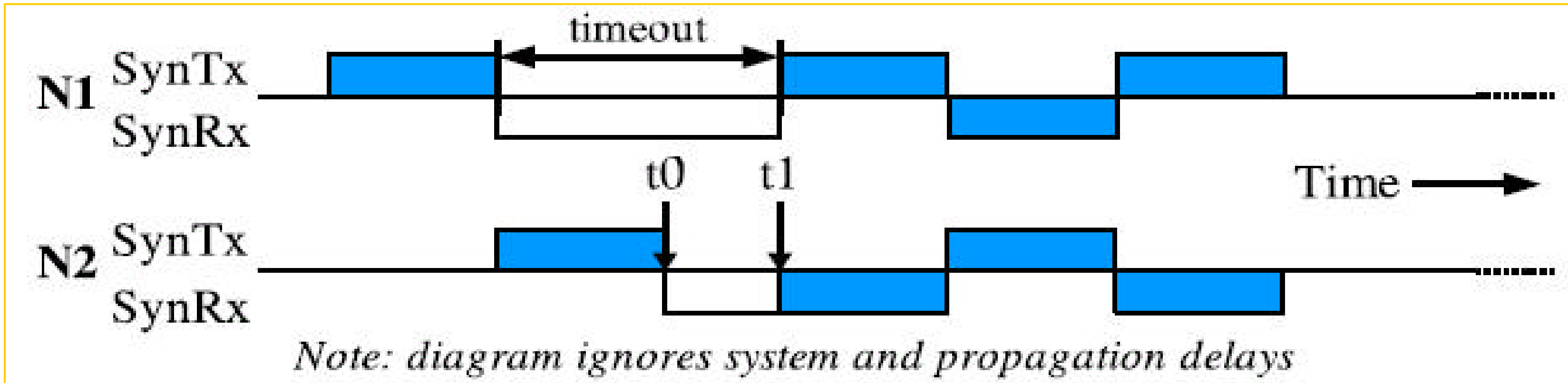


Figure 7. The 2P link-resync process

Solution: On entering SynRx, ifa starts a timer to control timeout

2P PROTOCOL DESIGN & IMPLEMENTATION (Cont.): Problems in Loose Synchrony

- **Two ends of a link get out of synchrony and timeout at the same time**

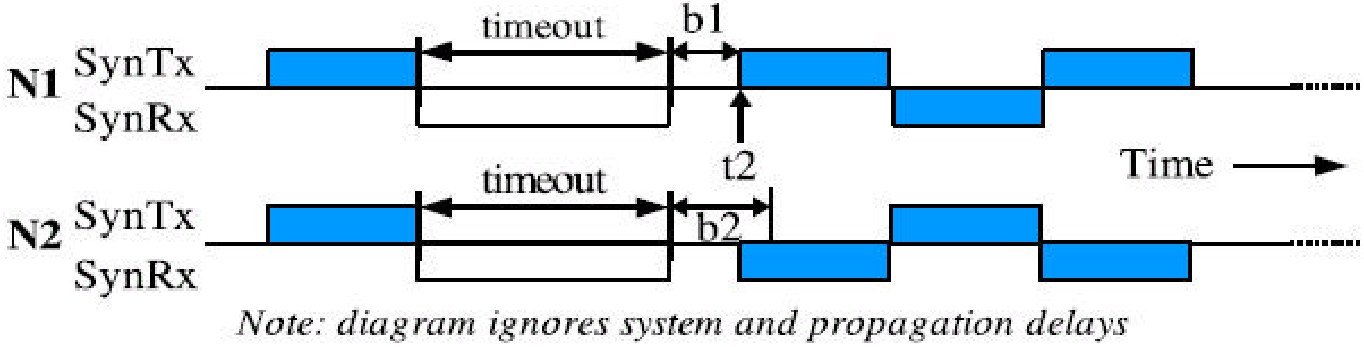


Figure 8. Bumping to avoid repeated timeouts

Solution: Add random perturbation (bumping) to the timeout value each time

2P PROTOCOL DESIGN & IMPLEMENTATION (Cont.): Communication Across Interfaces

- **Coordination of interfaces to switch from SynRx to SynTx**
  - Once an ifa decides to switch to Tx, it sends a notification (NOTIF) to other ifa-nbrs', and waits for NOTIF from them.
  - Aware of UP / DOWN status of other ifa-nbrs'. (observation of 3 consecutive time-outs implies DOWN)
- **Coordination of interfaces to switch from SynTx to SynRx**
  - Not necessary since all ifas' begin Tx simultaneously and with the same duration of B bytes

OUTLINE

- Introduction
- Background
- Protocol Design and Implementation
- Topology Construction
- Evaluation
- Discussion and Conclusions
- Comments

TOPOLOGY CONSTRUCTION

- **Constraints in Topology**
  - **Bipartite Constraint:**
    - If a node is in SynRx its neighbors should be in SynTx and vice versa
    - Implies no odd cycles are present
  - **Power Constraint: For proper reception we require that**
    - the signal level is above min. reqd. power level P_{min}
    - SINR has to be above the interference by SIR_{reqd}

TOPOLOGY CONSTRUCTION (Cont.):

- **For a given topology**
  - Power transmission P_i's, $(i = 1,2,\ldots N_A)$ are variables
  - d(i, j), distance between the nodes corresponding to antennae a_i and a_j is known
  - g(i, j), effective gain when a_i is transmitting and a_j is receiving, is known

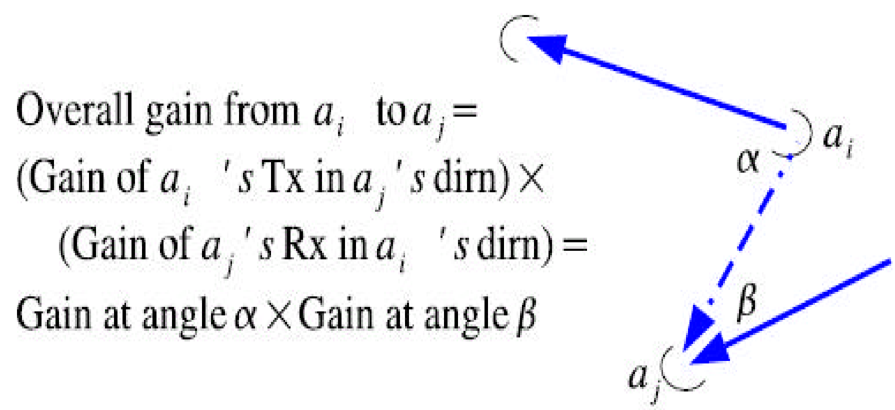


Figure 9. Illustrating gain from a_i to a_j

TOPOLOGY CONSTRUCTION (Cont.): Power Equations

$$\frac{P_{2k} \times g(2k, 2k-1)}{PL[d(2k, 2k-1)]} \geq P_{min} \qquad (1)$$

← Transmission power

$$Interf(2k-1) = \sum_{j=1, j \epsilon I(2k)}^{N_A} \frac{P_j \times g(j, 2k-1)}{PL[d(j, 2k-1)]} \qquad (2)$$

$$\frac{P_{2k} \times g(2k, 2k-1)}{PL[d(2k, 2k-1)]} \geq SIR_{reqd} \times Interf(2k-1) \quad (3)$$

} Considered as interference from all other nodes

Eq. 1 and 3 are power equations.

TOPOLOGY CONSTRUCTION (Cont.): Parameters in the Power Equations

- P_min: -85 dB for 11Mbps reception
- SIR_reqd: 10 dB for the 10^{-6} BER level, set to 14-16 dB in topology construction
- The antenna radiation pattern that decides the gain in different angles.

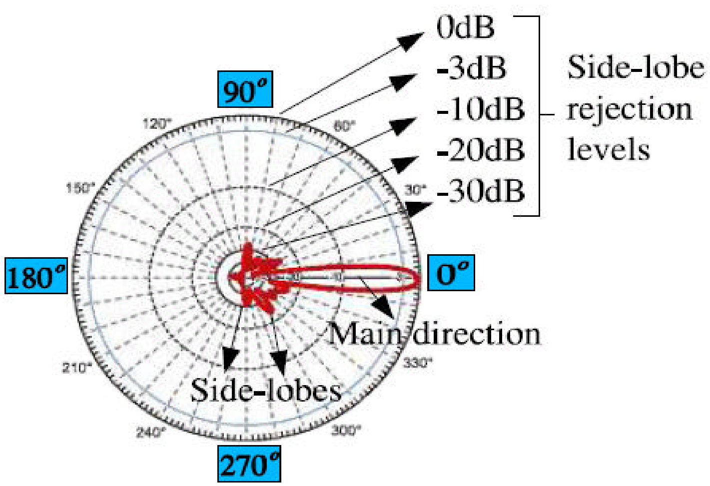


Figure 10. Spatial radn. pattern: parabolic grid antenna

TOPOLOGY CONSTRUCTION (Cont.): Topology Formation

- **Construct a tree topology that satisfies the two constraints**
  - **Suppose all (or most) traffic passes through the land-line node and don’t do multi-path routing**
  - **A tree rooted at the land-line node satisfies the bipartite constraint**
  - **Fault tolerance can be solved by morphing**

TOPOLOGY CONSTRUCTION (Cont.): Topology Formation

- **Form a spanning tree with following heuristics**
  - (H1) Reduce length of links used
    - Interference and power consumption
  - (H2) Avoid “short” angles between links
    - Side-lobe leakage
    - *ang_thr* of 30 to 45 degrees
  - (H3) Reduce hop-count
    - Deep trees = bad latency

TOPOLOGY CONSTRUCTION (Cont.): Algorithm

1. Set of Unconnected nodes is U, set of all possible connection links is S, create links at h_i
2. Order the links in S in increasing order of distance
3. For each link do
   - angle threshold check: ignore if angle < ang_thr, else add
   - Feasibility check (power constraint equation)
4. If all nodes connected, stop.
5. If successful in adding link in step 3, continue with step 1
6. If not successful in adding link in step 3, and link formed in h_i, go to next link, go to step 1.
7. If not successful in adding any link, and no link formed for h_i, declare failure, and stop.

OUTLINE

- Introduction
- Background
- Protocol Design and Implementation
- Topology Construction
- Evaluation
- Discussion and Conclusions
- Comments

EVALUATION: of topology creation

- **Purpose**
  - The effectiveness of the algorithm
  - The effect of varying the parameter SIR_{reqd}
- **Evaluation subjects**
  - 4 collections of villages from a local district map
    - Q1, Q2, Q3 and Q4
    - Q1 has 31 nodes
    - Q2-Q4 have 32 nodes, respectively
  - Topologies randomly generated
    - 50 nodes in an area of 44Km X 44Km

EVALUATION: of topology creation

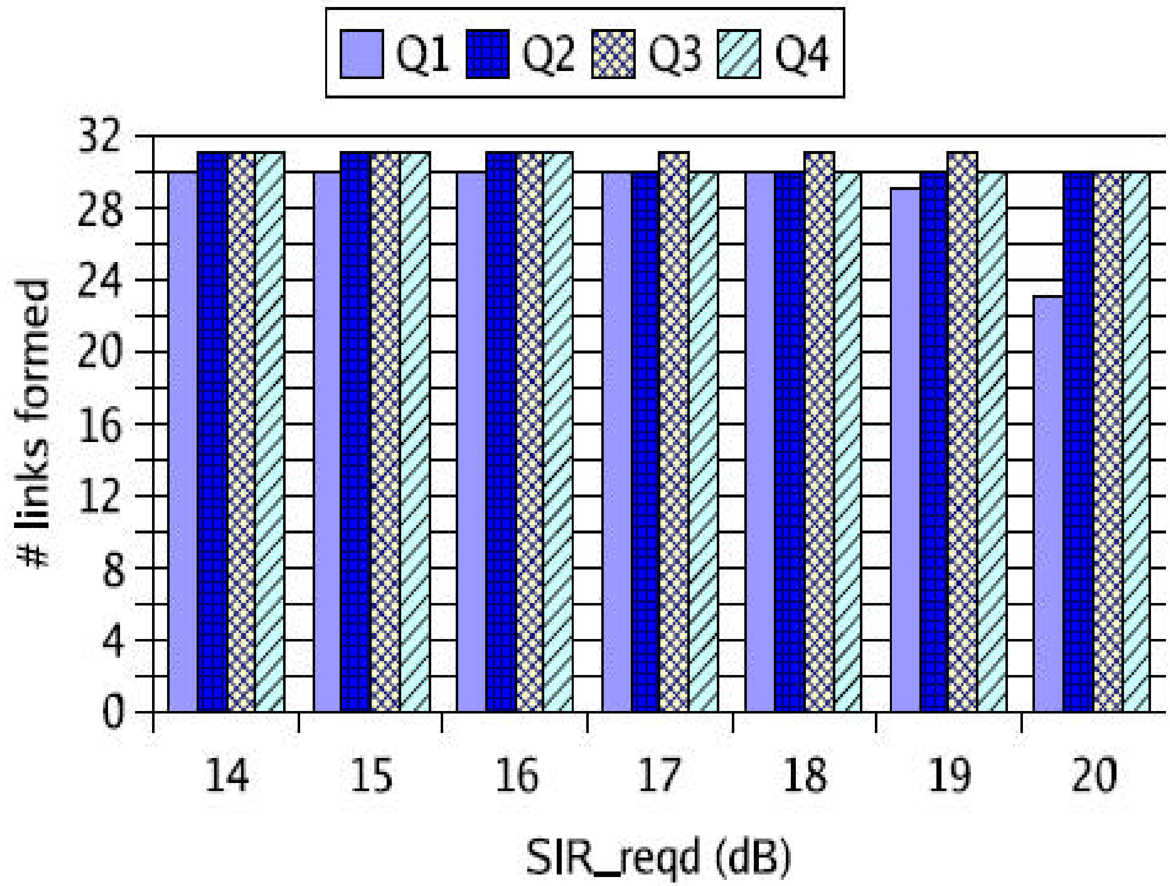


Figure 11. Topology formation on Q_1, Q_2, Q_3, and Q_4

EVALUATION: of topology creation

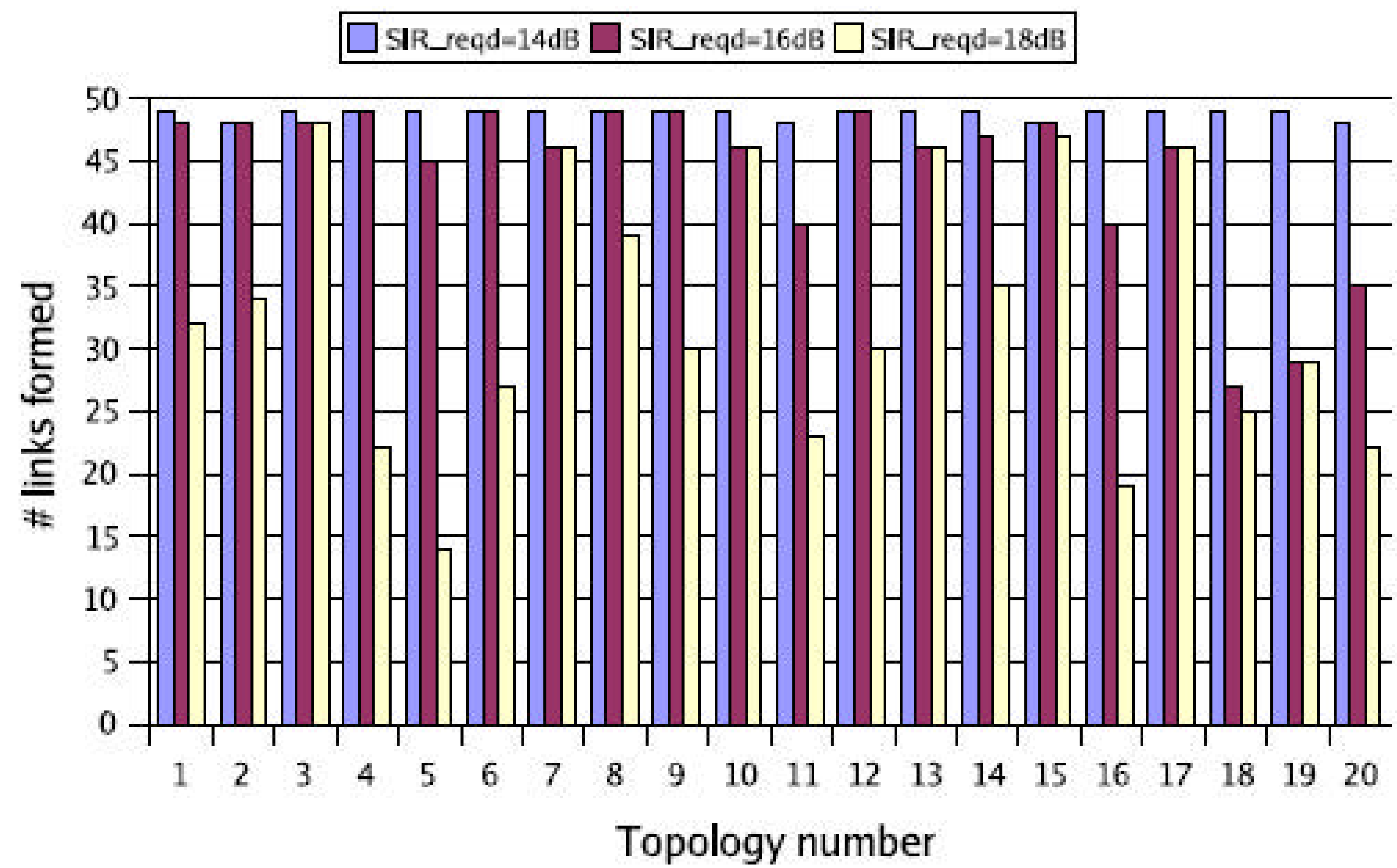


Figure 13. Topology formation on random 50-node cases

EVALUATION: of topology creation

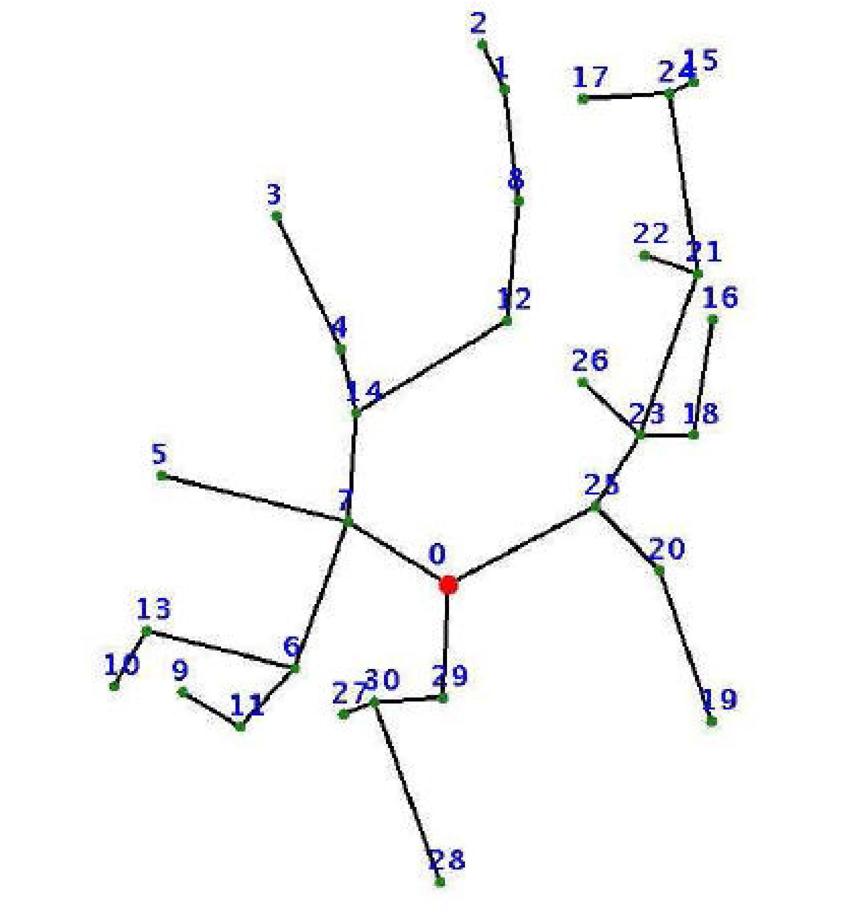


Figure 12. An example tree created on Q_1

EVALUATION: simulation studies

- **Goals:**
  - **To measure the impact that step by step link establishment has on loosely synchronized network**
  - **Saturation throughput performance compared to CSMA/CA protocol**
  - **Performance of TCP over 2P operated networks**

EVALUATION: extensions to ns-2

- **ns-2 extended for:**
  - **Multiple interface support**
  - **Directional antenna support**
  - **MAC modifications**
  - **LLC modifications**

EVALUATION: Simulation results

- **Link Establishment:**
  - **Method:** add links one after another to an already synchronized network
  - **Results:**
    - Took 12.9ms for first link establishment
    - Reason: first transmission of both ends of link coincide and had to use bumping to establish link
    - Took 4.9ms for rest of the links to establish
    - No noticeable difference in throughput of already synchronized links while adding new links

EVALUATION: Simulation results

- **Saturation throughput**
  - **UDP traffic**
    - One packet every 2ms
    - Packet size: 1400 bytes
  - **Results:**
    - Nodes operated in 2P achieve around 3-4 times more bandwidth than operated in the CSMA/CA protocol

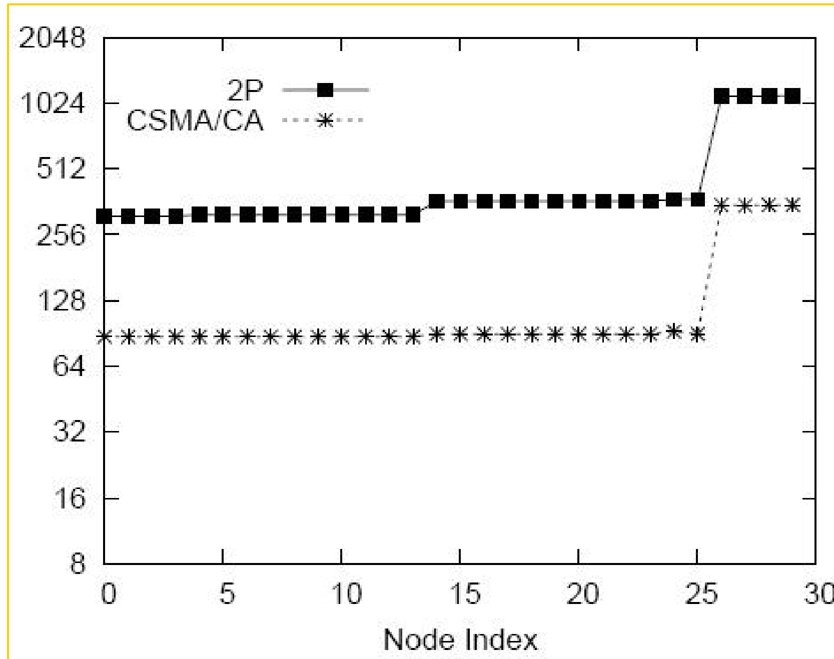


Figure 14. Saturation Throughput

EVALUATION: Simulation results

- **TCP Performance**
  - **In loss free: Up to 20 times better performance than CSMA/CA**

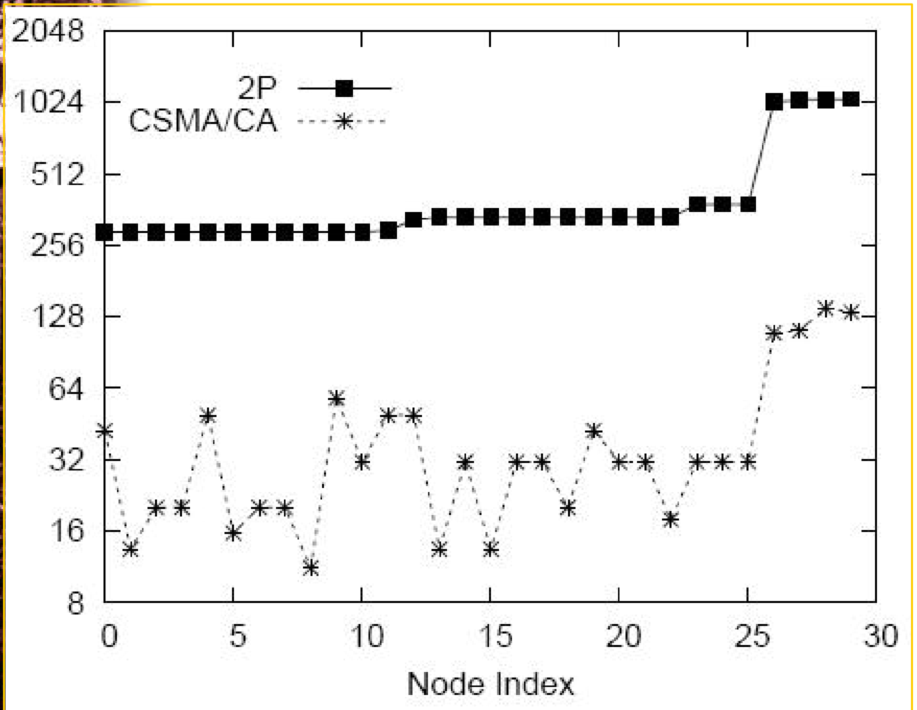
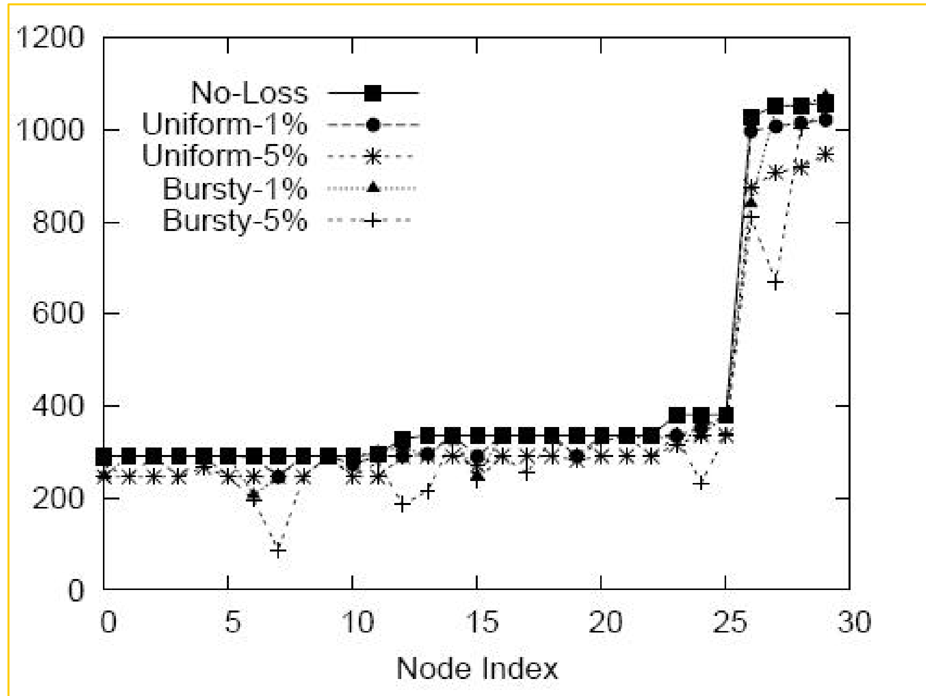


Figure 15. TCP Performance: No Losses



16. TCP Performance: Uniform and Bursty Losses

EVALUATION: Implementation based results

- **Prototype implementation on HostAP v0.2.4 on Linux v2.4.20-8**
- **Confirmation of SynOp with Prism2 cards:**
  - 6.5Mbps throughput on each link at the same time.
- **2P performance on a single link:**
  - 3.05Mbps average throughput – lower than 4.4Mbps observed in simulations
  - Overheads of marker pkts. And changing of antsel_rx in Prism2 cards give a combined throughput of 6.1Mbps which is less than 6.5Mbps observed.

EVALUATION: Implementation based results (Cont.)

- **Sub-optimal performance of 2P on a pair of links:**
  - Per interface throughput is lower than 3.05 Mbps because contention window set at 32 instead of 1 hence random backoff even in the absence of carrier sense
  - Limitations in driver level approach to 2P implementation
  - Stress of CPU scheduling involved in copying of rx/tx bytes to/from hardware as PCMCIA cards used didn't have Direct Memory Access

	Avg (SD) thrpt at A (Mbps)	Avg (SD) thrpt at N_1 (Mbps)	Avg (SD) thrpt at N_2 (Mbps)	Avg (SD) thrpt at B (Mbps)
2P	2.70 (0.31)	2.06 (0.24)	2.81 (0.15)	2.81 (0.10)
CSMA	2.07 (0.13)	1.13 (0.22)	1.90 (0.15)	3.11 (0.14)

Table 1. 2P on two links, versus CSMA

OUTLINE

- Introduction
- Background
- Protocol Design and Implementation
- Topology Construction
- Evaluation
- Discussion and Conclusions
- Comments

Discussion and Conclusions

- **Prior work involves Spatial reuse Time Division Multiple Access (STDMA) scheduling**
- **The present work differs in:**
  - ✓ Multiple radios per node
  - ✓ Directional antennae
  - ✓ Exact location of nodes
- **Fault tolerance and Morphing**
  - ➢ Trees are not very fault tolerant
  - ➢ Morph the topology in the event of a failure
    - -Provision additional links, but turn them on only as needed
  - ➢ Morphing can be used to create new routes when network equipment is turned off

OUTLINE

- Introduction
- Background
- Protocol Design and Implementation
- Topology Construction
- Evaluation
- Discussion and Conclusions
- Comments

COMMENTS

Pros:

1. Performance enhancement
2. Low cost implementation
3. Fault tolerance solution
4. Feasible protocol

Cons:

1. Requires one dedicated transceiver for each link
2. Reconfigure on node’s joining / removal / relocation
3. Topology is centralized with multiple landlines
4. Transmit empty pkts – fairness & security

QUESTIONS


ANY QUESTIONS?
HOW DO I TRANSFER TO THE HUMANITIES DEPARTMENT?
8-30
THAVES