

# **On the Design and Performance of Cognitive Packets over Wired Networks and Mobile Ad Hoc Networks**

## **Dissertation Defense**

M. Ricardo Lent

School of Electrical Engineering and Computer Science  
University of Central Florida

March 3, 2003

# Introduction

- Routing allows communication of non-adjacent nodes
- *Routing algorithms* dynamically construct paths that are stored in *routing tables* (RT)
- Current and future applications require paths that match a QoS level
- Many previous solutions require periodic RT exchange and central computation of routes on a single criterion
- Cognitive Packet Networks (CPN). A new routing approach:
  - Emphasis on learning
  - Employ focalized information collection
  - Users and network requirements define the QoS of each connection

# Main Contributions

- Completed design of CPN\* routing algorithm
  - \* *Towards Networks with Intelligent Packets*. Gelenbe, Seref, Xu. Proceedings of IEEE-ICTAI Conference on Tools for Artificial Intelligence, November 1999
- Extended CPN to MANET (propagation mechanism, neighboring discovery, adequate routing goal)
- Defined and implemented CPN and AHCPN protocols, and deployed network testbeds
- Verified the adaptability capabilities of CPN and AHCPN (measurements and simulation)

# Agenda

- Introduction to cognitive packet networks (CPN)
- Protocol implementation
- Performance measurements
- Ad hoc cognitive packet networks (AHCPN)
- Simulation results
- Summary and conclusions

# Cognitive Packet Networks

Cognitive packet networks (CPN) are packet switching networks that implement adaptive, QoS-driven routing

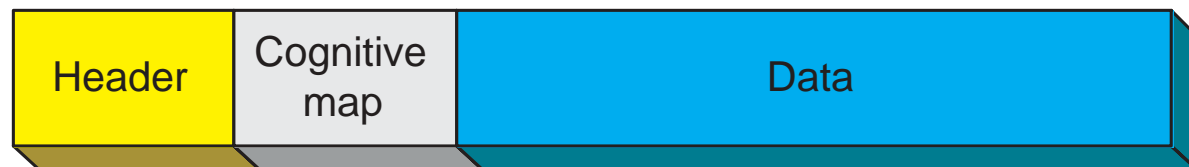
## Type of packets:

### *Data Plane:*

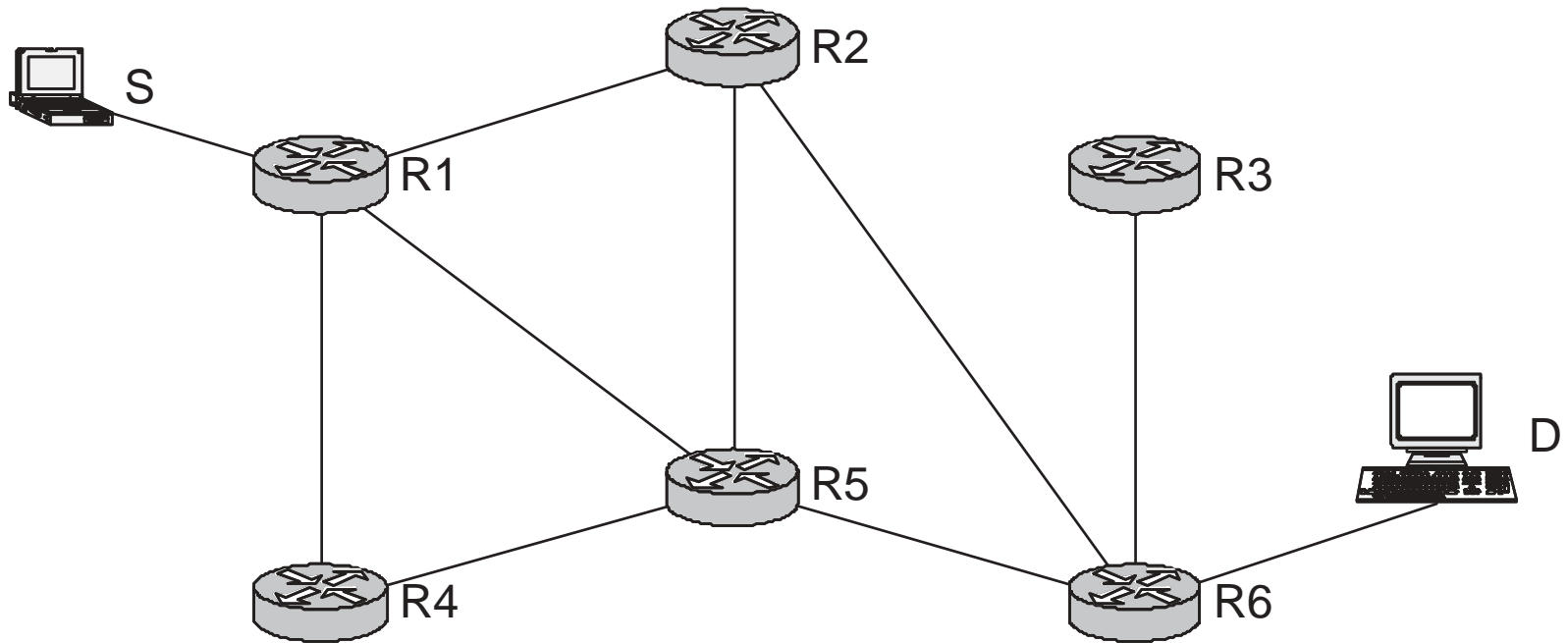
- **DUMB Packets** carry payload (source routing)

### *Control Plane:*

- **SMART or Cognitive Packets** use Random Neural Networks with a Reinforcement Learning algorithm to take routing decisions
- **ACK Packets** distribute routing information (source routing)

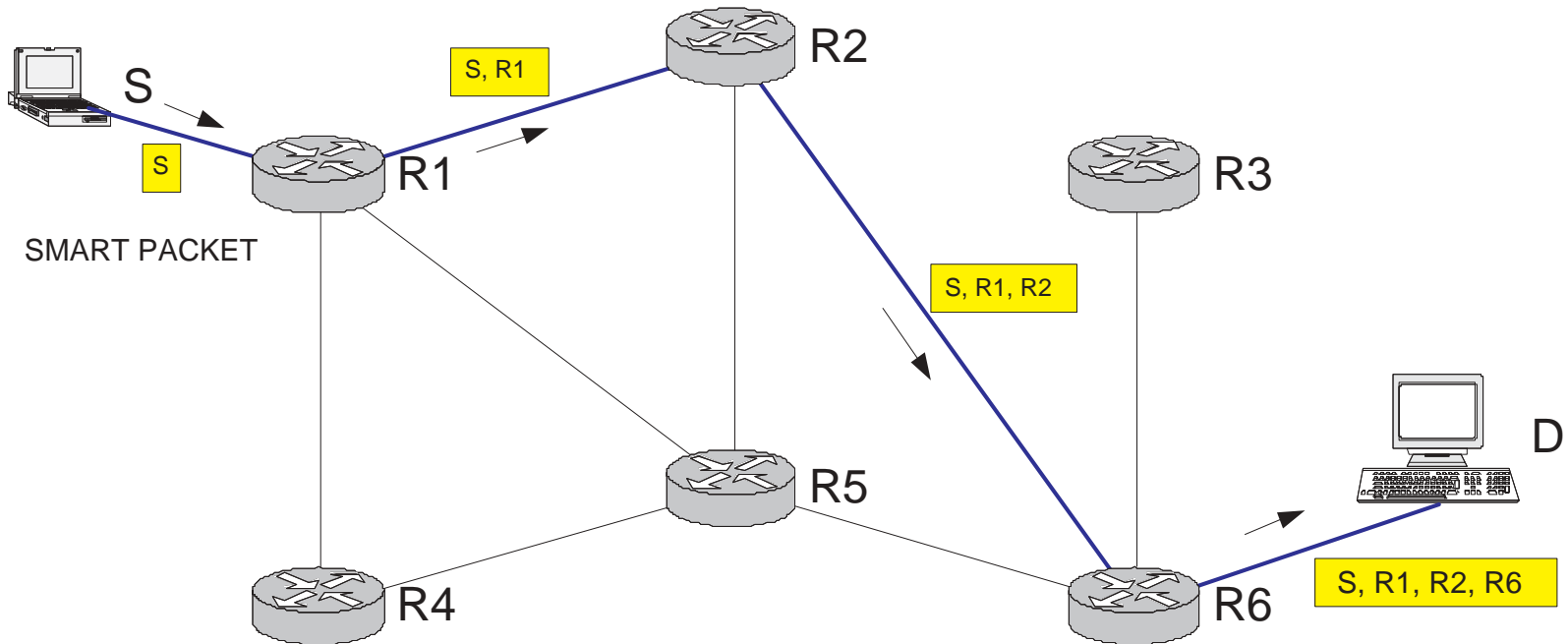
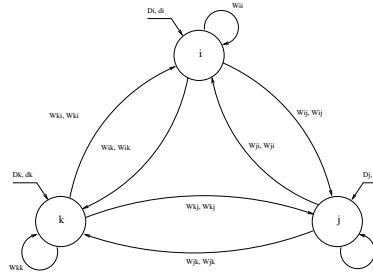


# CPN Routing



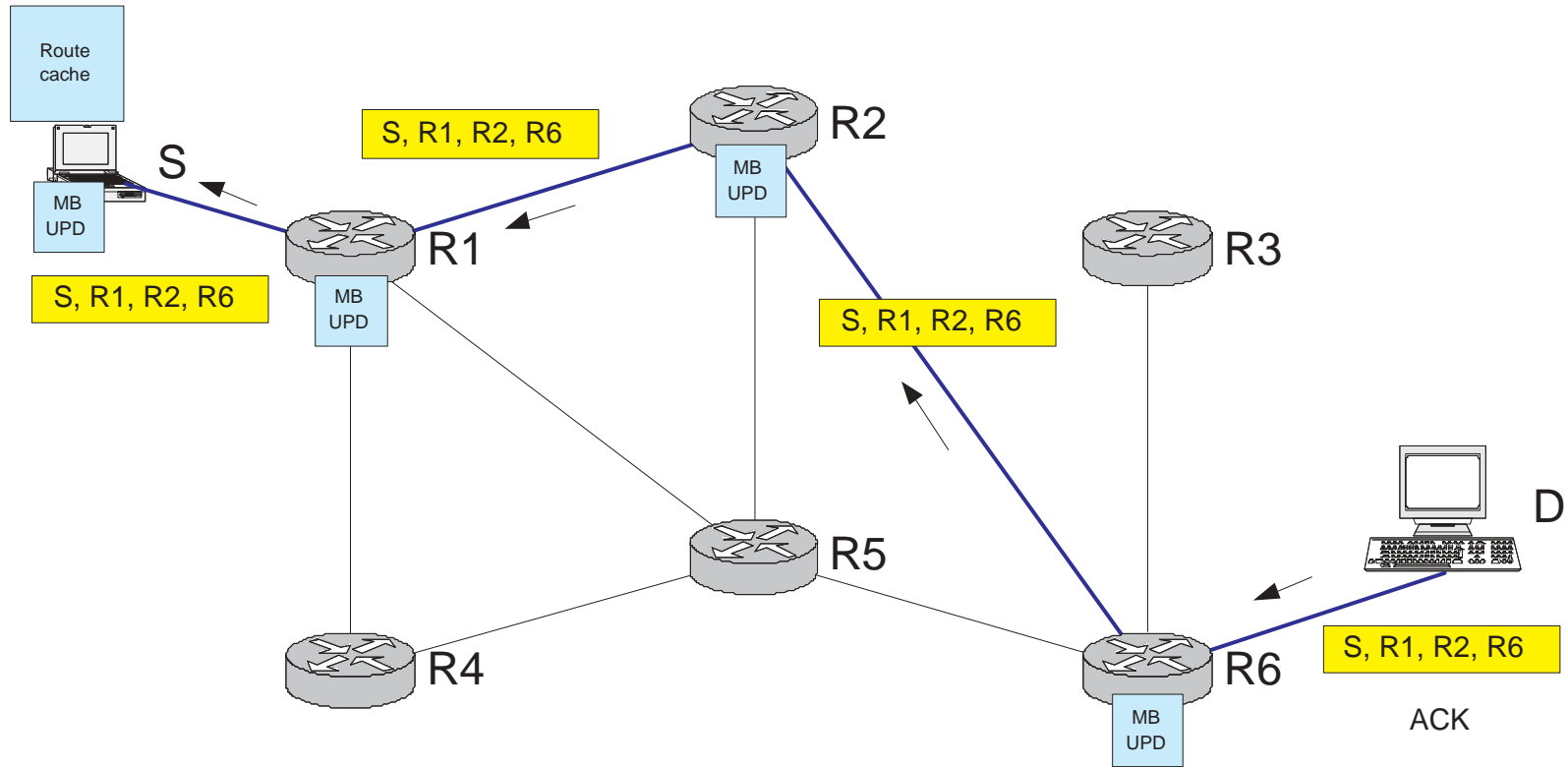
Routing goal = minimization of round-trip delay

# CPN Routing (cont'd)



- *Cognitive map* stores addresses and timestamps
- Potential loops are removed (node by node or at destination)

# CPN Routing (cont'd)



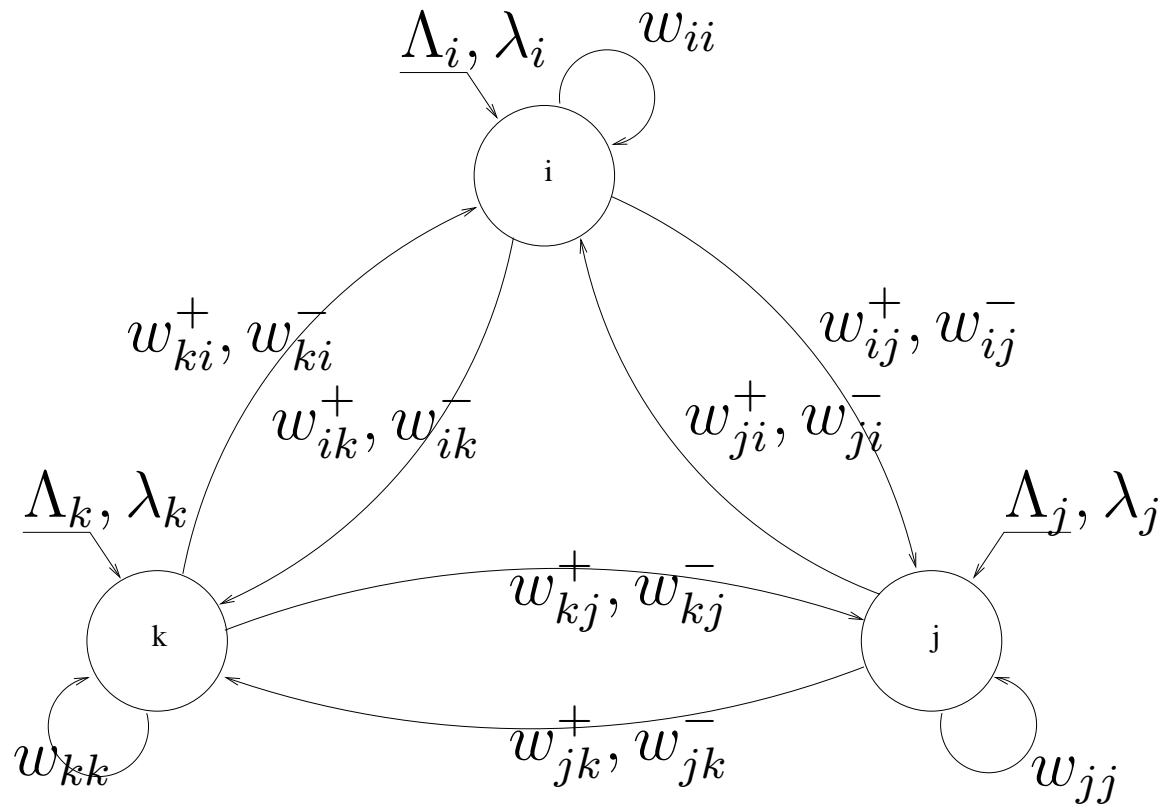
- ACK updates *Mailboxes* (average RTT to destination)
- ACK creates or updates route in *route cache*
- DP uses source routing from information in *route cache*
- An SP ratio ensures route maintenance

# Random Neural Networks\*

- Mathematical structure similar to a queuing network
- Fully connected network of  $N$  neurons
- Neuron  $i$  is characterized by a potential level  $q_i$  (*excitation level*)
- Neurons fire unit-amplitude spikes
- Spikes create excitatory or inhibitory signals to other neurons
- Always converge to steady state in finite time

\* *Spiked Random Neural Networks*, E. Gelenbe, Neural Computation 93

# RNN example



$w_{ij}^+ = r(i)p^+(i, j)$  and  $w_{ij}^- = r(i)p^-(i, j)$

are the rates at which neuron  $i$  sends excitation and inhibition spikes to neuron  $j$  (when  $i$  is excited) respectively

# Steady-state Equations

Probability that the neuron is excited:

$$q_i = \frac{\lambda^+(i)}{r(i) + \lambda^-(i)}$$

$$\lambda^+(i) = \sum_j q_j w_{ji}^+ + \Lambda_i, \quad \lambda^-(i) = \sum_j q_j w_{ji}^- + \lambda_i$$

In CPN, the largest  $q_i$  selects the next-hop for the packet.

# Reinforcement Learning (RL)

Definitions:

- Goal to achieve  $G$  (example  $G = D$ )
- Reward function  $R = G^{-1}$
- A threshold function  $T_l$ :

$$T_l = \beta T_{l-1} + (1 - \beta) R_l \quad ; 0 < \beta < 1$$

# RL (cont'd)

Learning comes from past experience:

If  $T_{l+1} \leq R_{l+1}$

🟡  $w_{ji}^+ \leftarrow w_{ji}^+ + R_l,$  *(reward)*

🟡  $w_{ji}^- \leftarrow w_{ji}^- + \frac{R_l}{n-2}, \quad k \neq j,$

else

🟡  $w_{ji}^+ \leftarrow w_{ji}^+ + \frac{R_l}{n-2}, \quad k \neq j,$  *(punishment)*

🟡  $w_{ji}^- \leftarrow w_{ji}^- + R_l.$

$R_l$  is calculated from information stored in Mailbox

# Normalization

$$r_i^* = \sum_1^n [w^+(i, m) + w^-(i, m)] \quad ; i = 1, \dots, n,$$

$$w^+(i, j) \leftarrow w^+(i, j) * \frac{r_i}{r_i^*}$$

$$w^-(i, j) \leftarrow w^-(i, j) * \frac{r_i}{r_i^*}$$

Prevents ever-growing values

# Agenda

- Introduction to cognitive packet networks (CPN)
- **Protocol implementation**
- Performance measurements
- Ad hoc cognitive packet networks (AHCPN)
- Simulation results
- Summary and conclusions

# Implementation Model

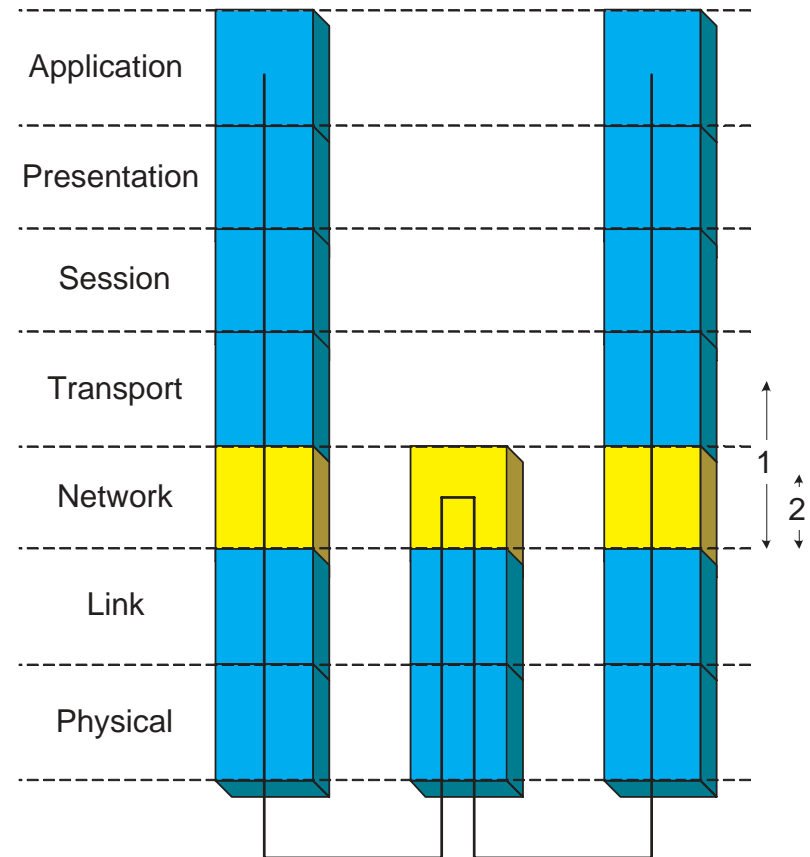
Linux 2.2 and 2.4 kernel

## 1. Network module

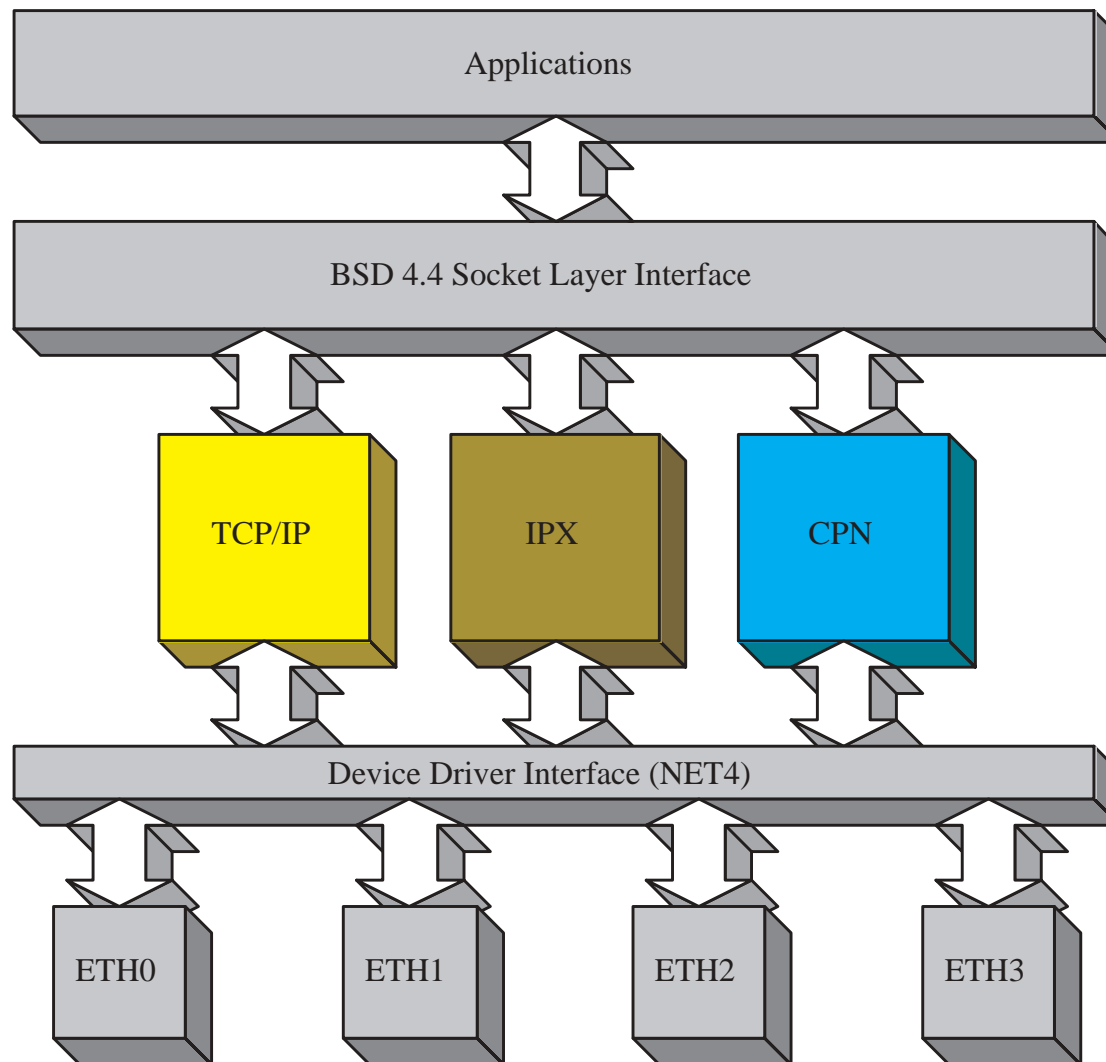
- Complete layer-3, partial layer-4 implementation ( UDP/IP)
- May use TCP/UDP
- Tunneling (IP)

## 2. Virtual interface

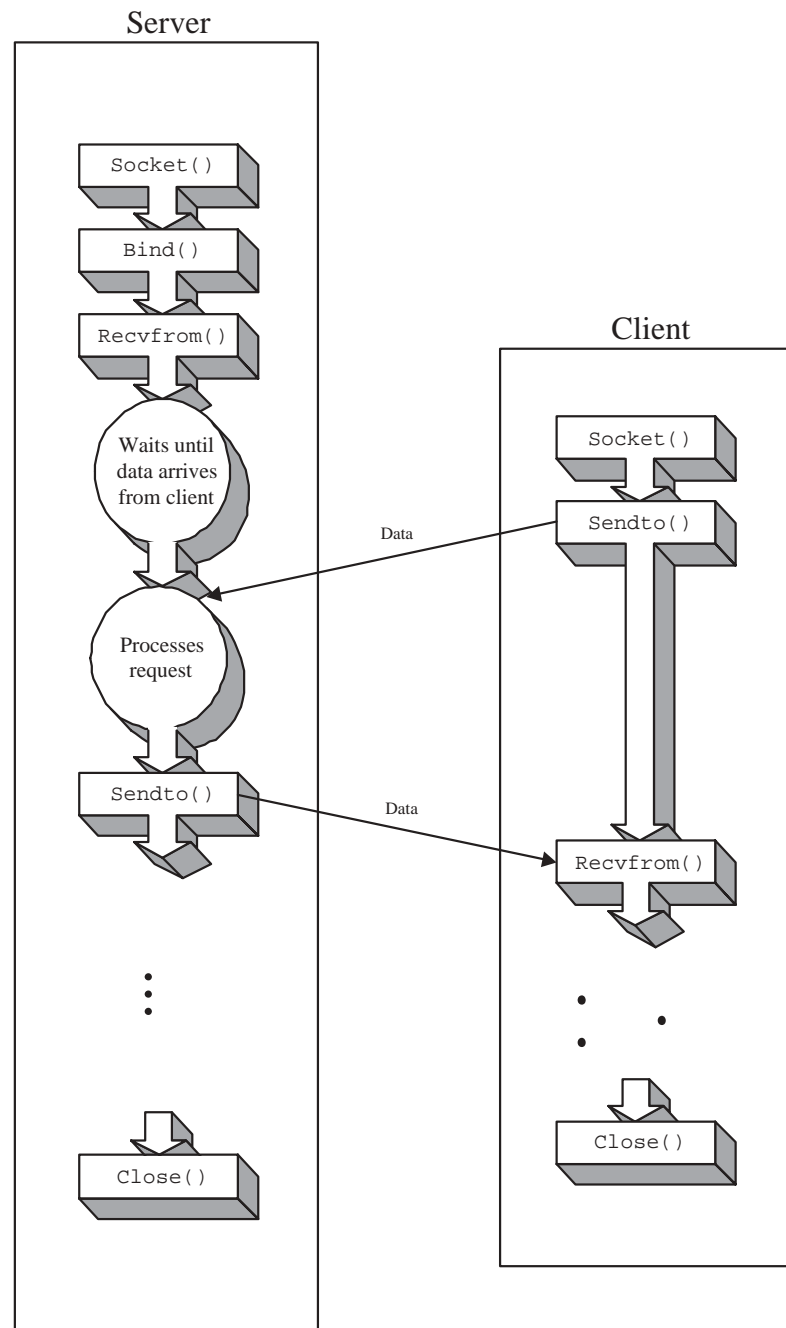
- Sub-Layer 3 implementation
- Seamless interoperation with IP



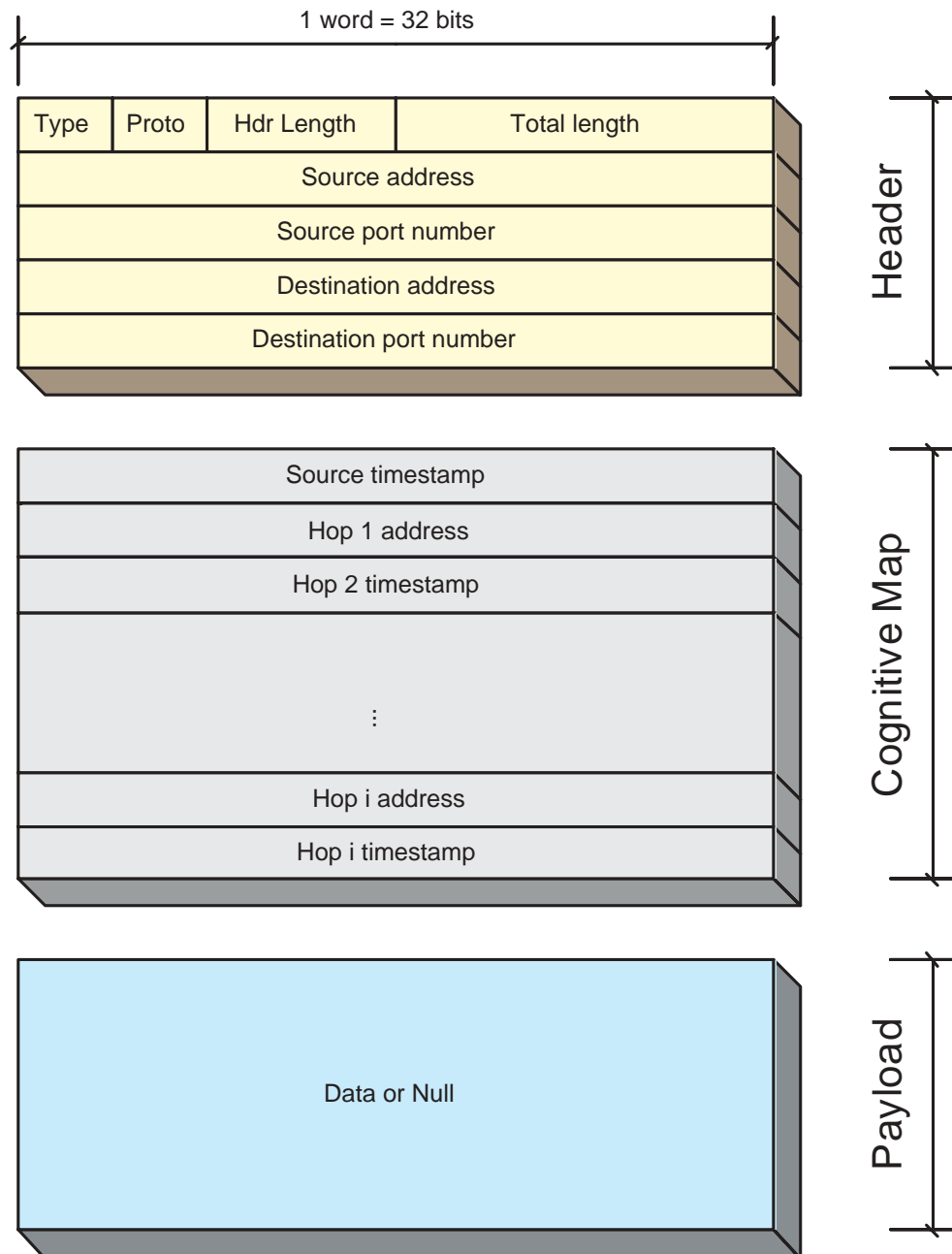
# A. Linux Kernel Module



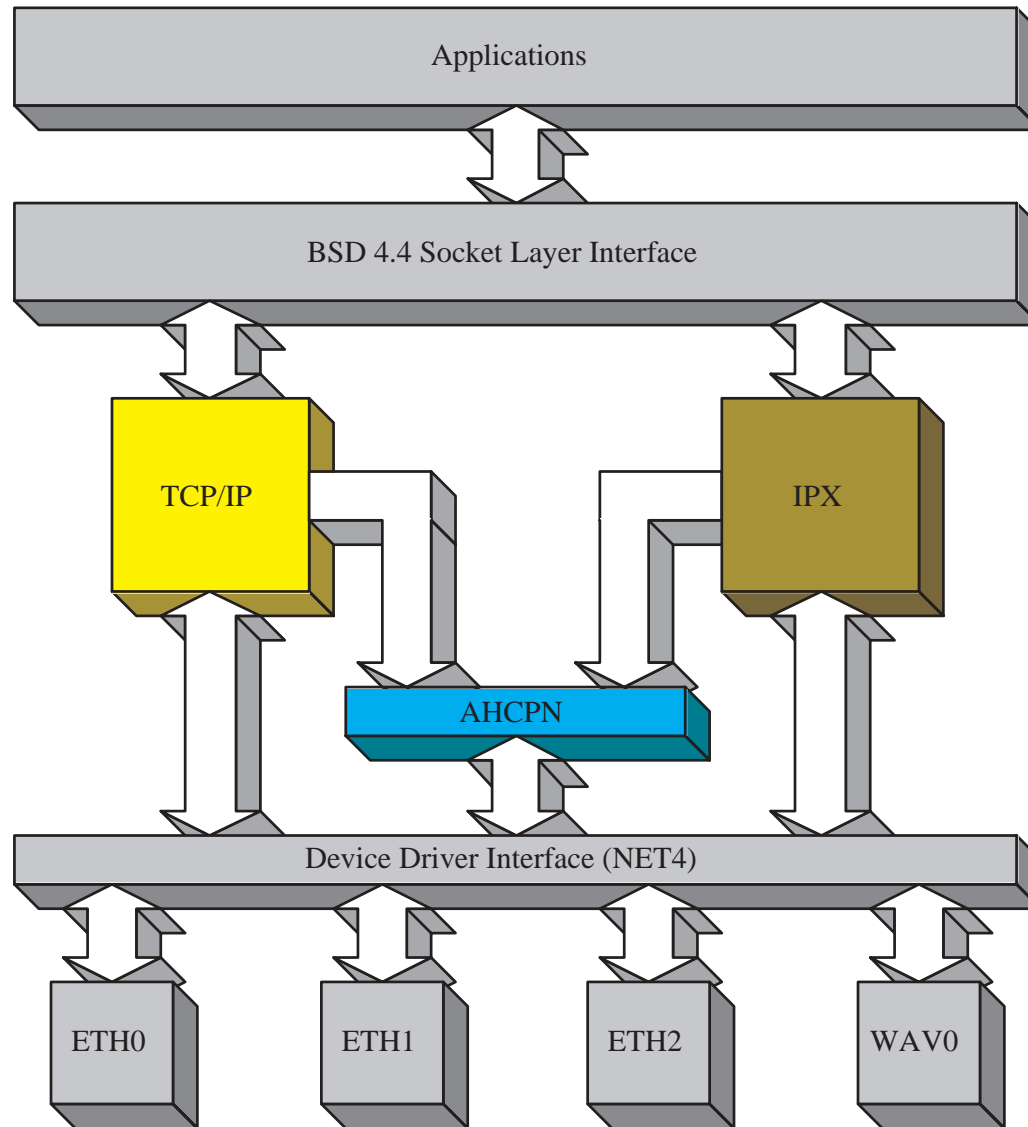
# Application Programming Interface



# Packet Structure 1



# B. Virtual Interface



# AHCPN and IP

```
eten - SecureCRT
File Edit View Options Transfer Script Window Help
[Icons]
eten:/# ifconfig cix0
cix0      Link encap:Ethernet  HWaddr 00:00:00:00:00:00
          inet addr:192.168.2.1  Mask:255.255.255.0
          UP RUNNING NOARP  MTU:1000  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

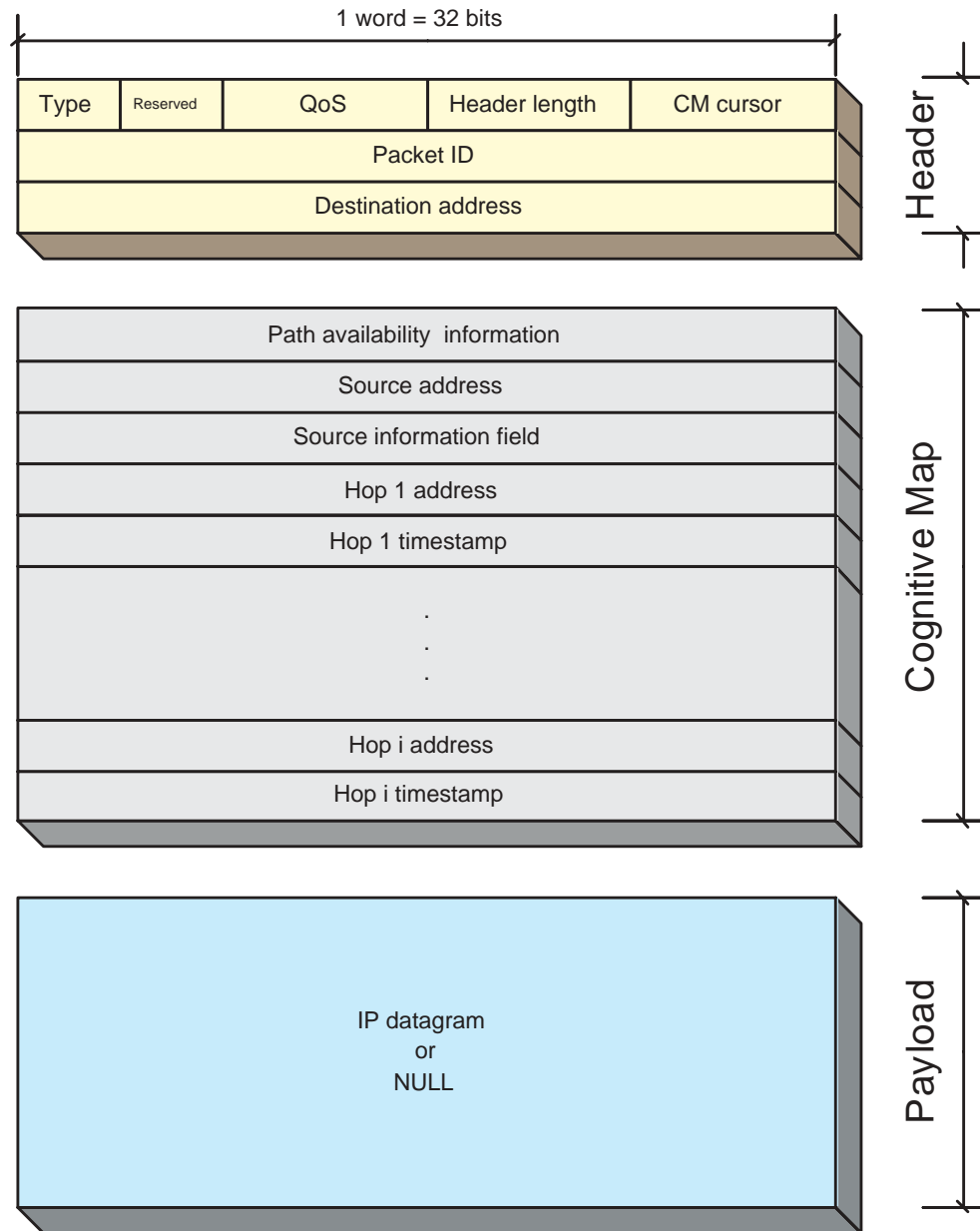
eten:/# netstat -i
Kernel Interface table
Iface  MTU Met  RX-OK RX-ERR RX-DRP RX-OVR   TX-OK TX-ERR TX-DRP TX-OVR Flg
cix0   1000  0     0     0     0     0     0     0     0     0  0 ORU
eth0   1500  0  13022680  0     0     1  221766  0     0     0  0 BMRU
eth1   1500  0  337785   0     0     0     0  398712  0     0     0  0 BMRU
eth2   1500  0     0     0     0     0     0     0     0     0  0 BMPRU
eth3   1500  0     0     0     0     0     0     0     0     0  0 BMPRU
lo     16436 0     79     0     0     0     79     0     0     0  0 LRU

eten:/# ping -c 3 192.168.2.1
PING 192.168.2.1 (192.168.2.1): 56 data bytes
64 bytes from 192.168.2.1: icmp_seq=0 ttl=255 time=0.2 ms
64 bytes from 192.168.2.1: icmp_seq=1 ttl=255 time=0.0 ms
64 bytes from 192.168.2.1: icmp_seq=2 ttl=255 time=0.0 ms

--- 192.168.2.1 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0.0/0.0/0.2 ms
eten:/#
```

Ready Telnet 28, 9 28 Rows, 80 Cols VT100 NUM

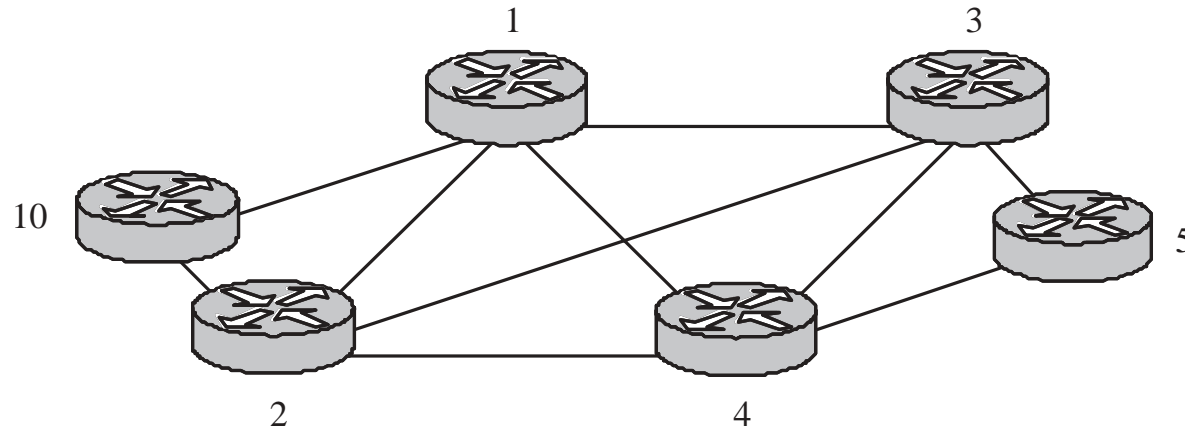
# Packet Structure 2



# Agenda

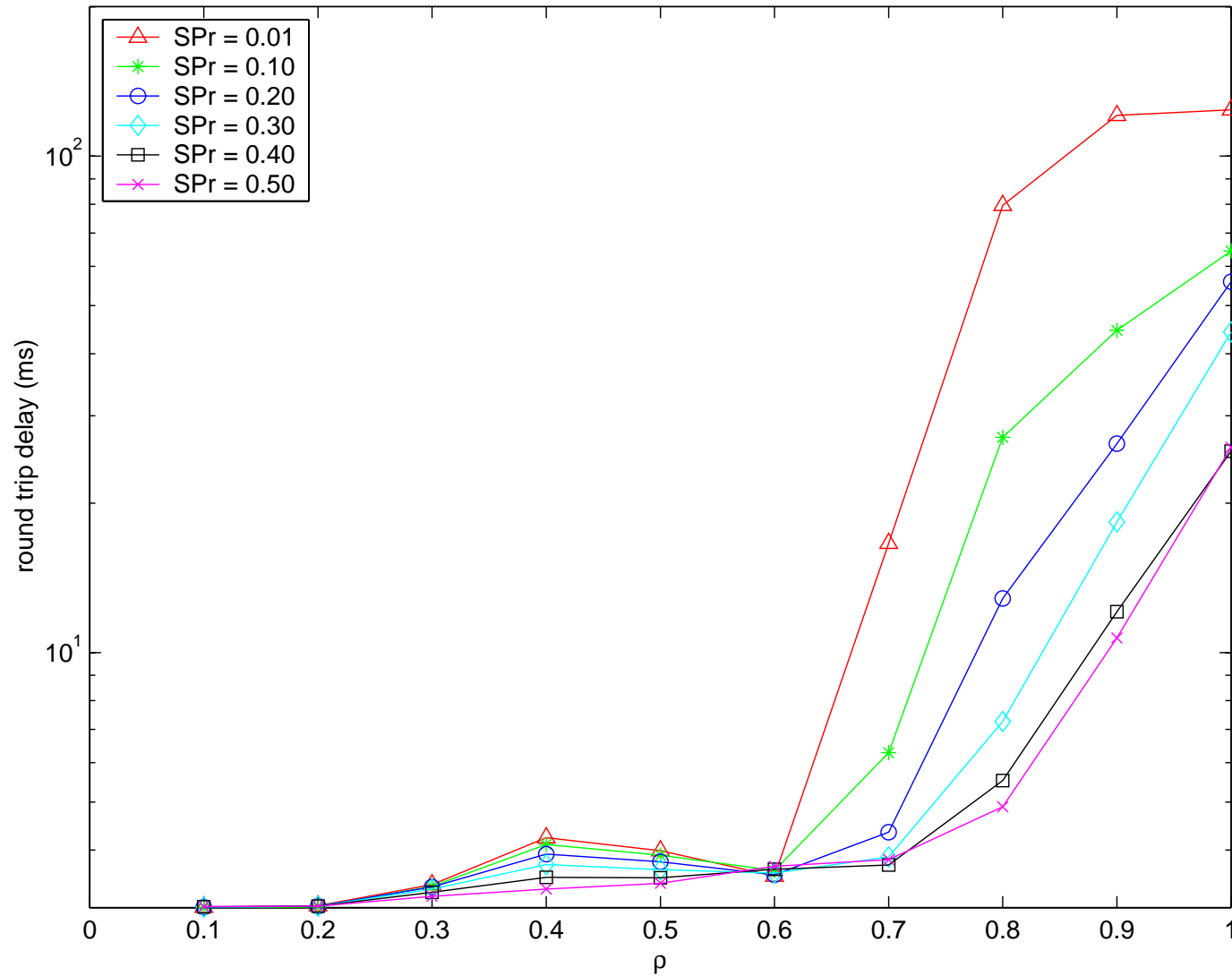
- Introduction to cognitive packet networks (CPN)
- Protocol implementation
- **Performance measurements**
- Ad hoc cognitive packet networks (AHCPN)
- Simulation results
- Summary and conclusions

# Test-bed Network Topology

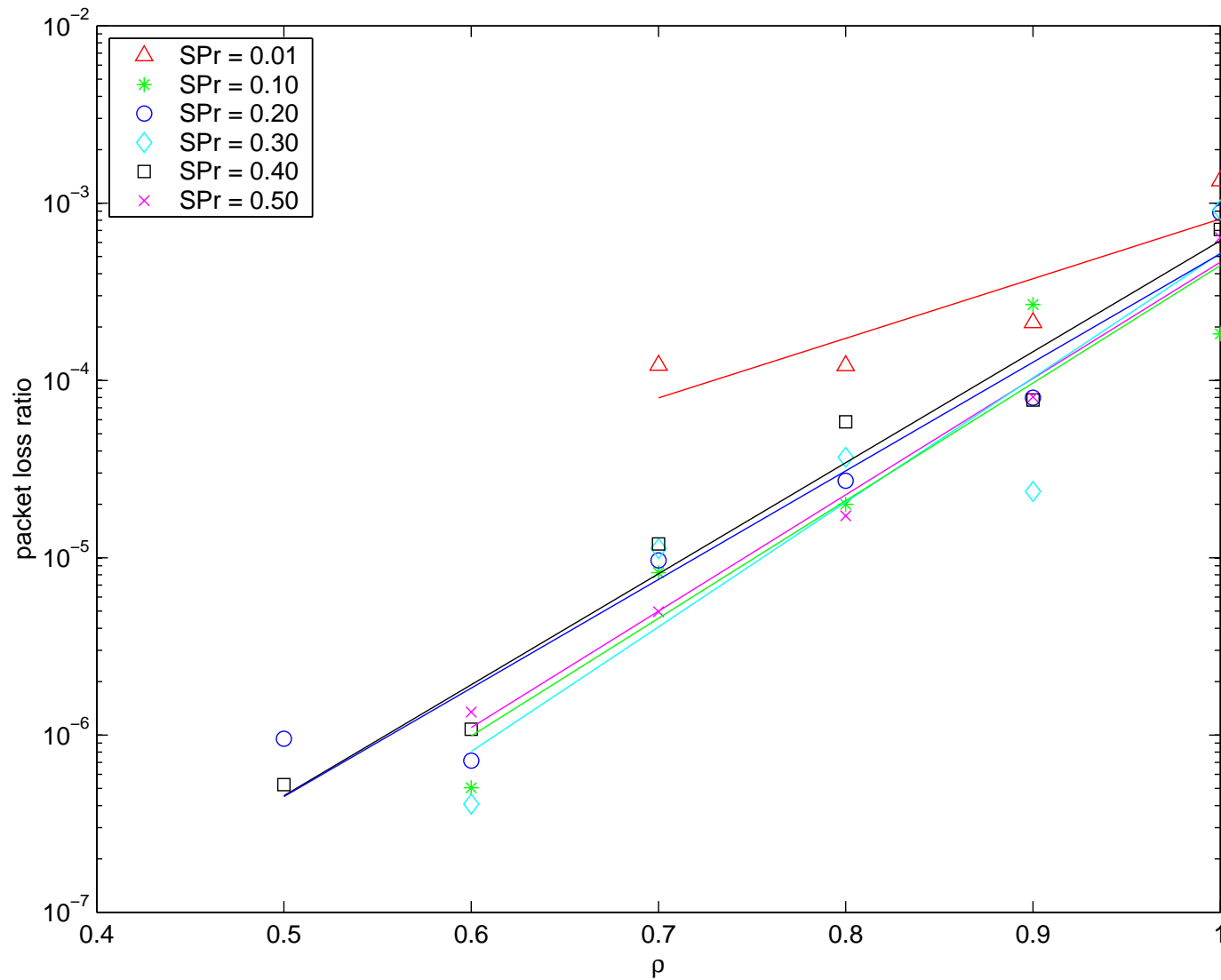


- PC based: Intel P3 450 MHz, 128 Mb RAM, 4 Ethernet cards (10Mbps)
- Test traffic: CBR from 10 to 5, packets of 1024 bytes
- Selected results on the adaptation capability of CPN:
  - under regular conditions
  - under failures

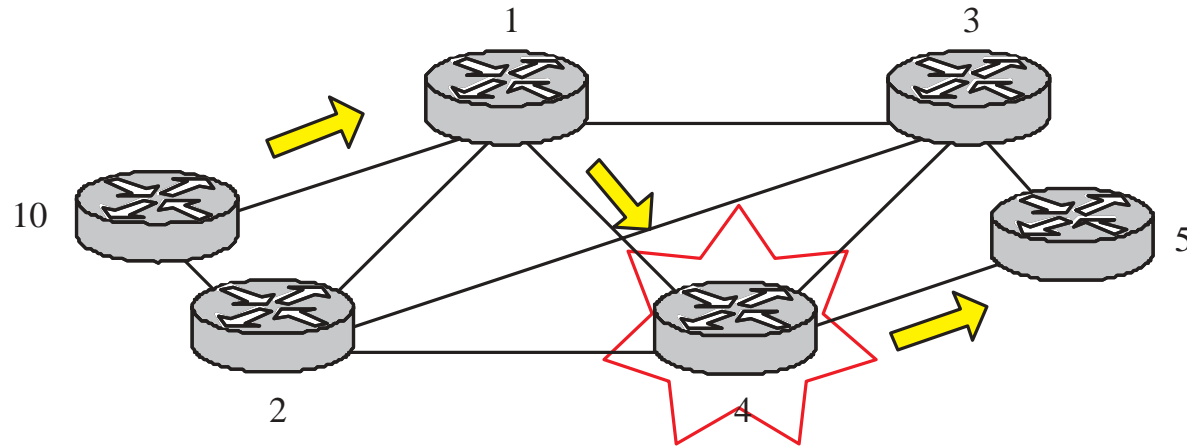
# Round-trip delay (DP) vs. Offered Load



# Packet Loss Ratio (DP) vs. Offered Load



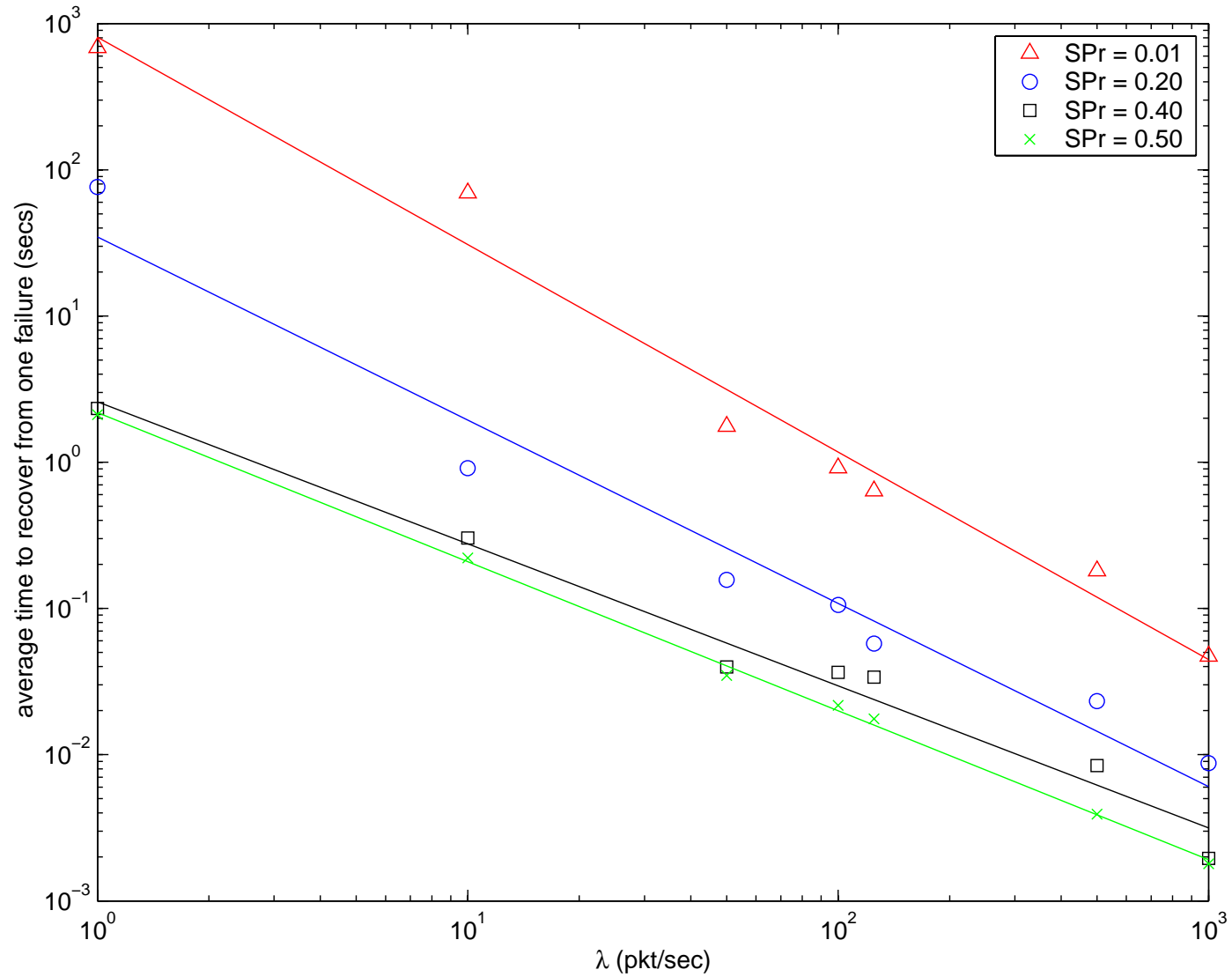
# Survivability Evaluation



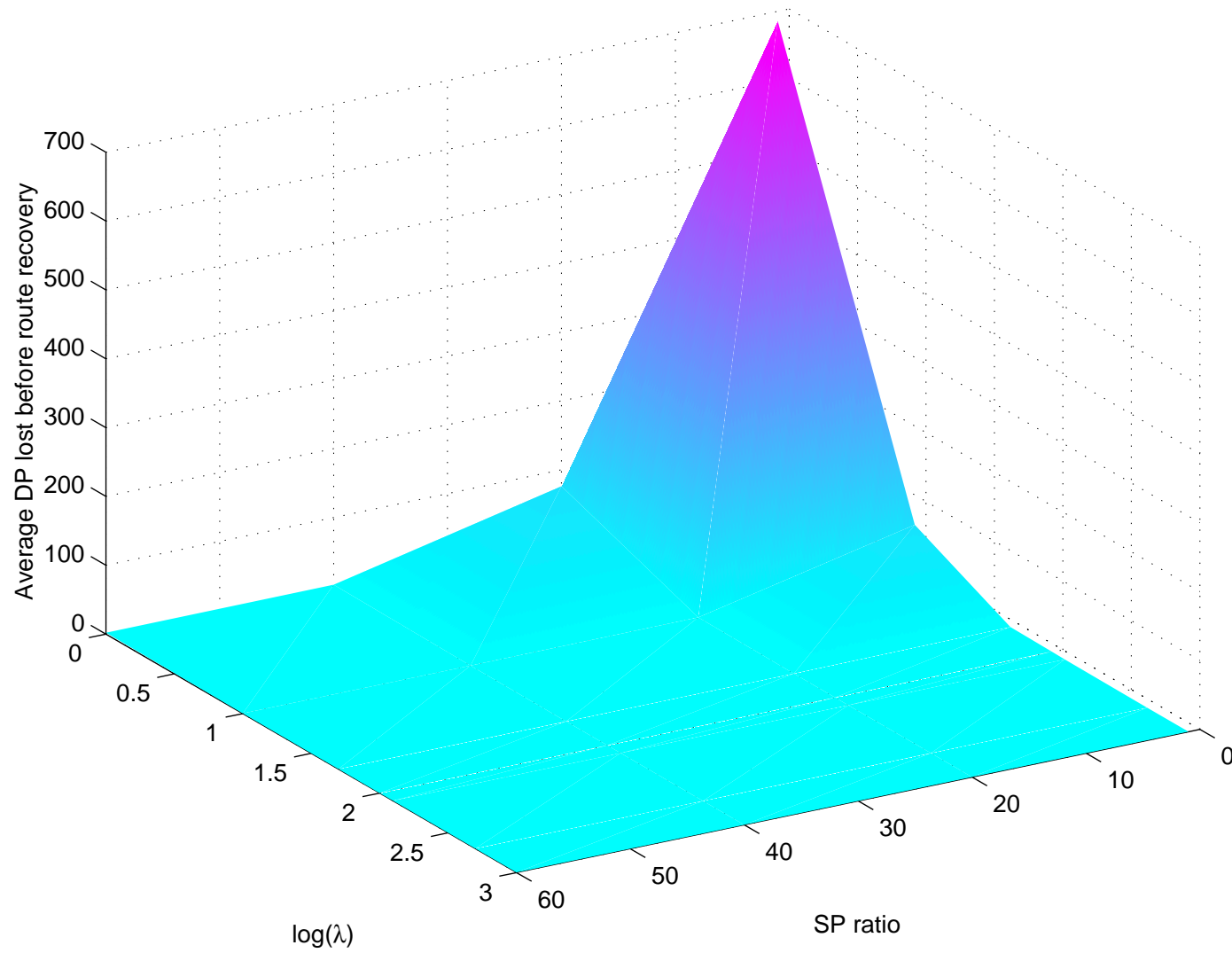
Performance measurements study:

- Average time to recover from a failure
- Resulting performance after failures

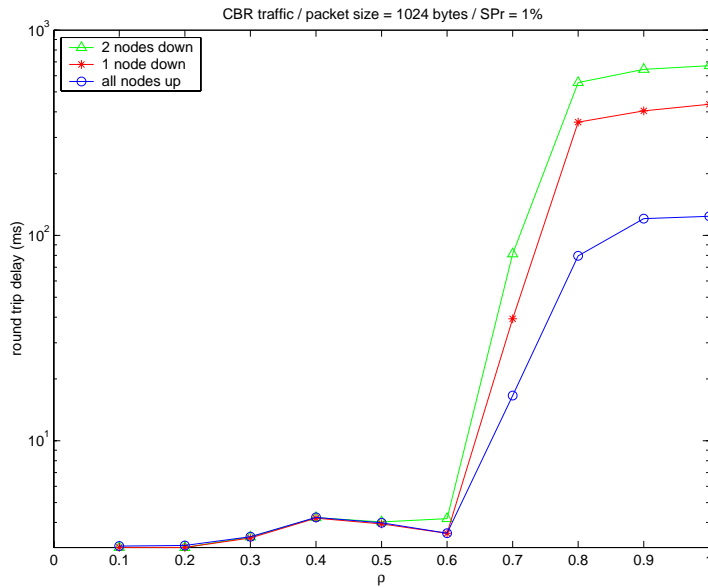
# Route Repair Time vs. Offered Load



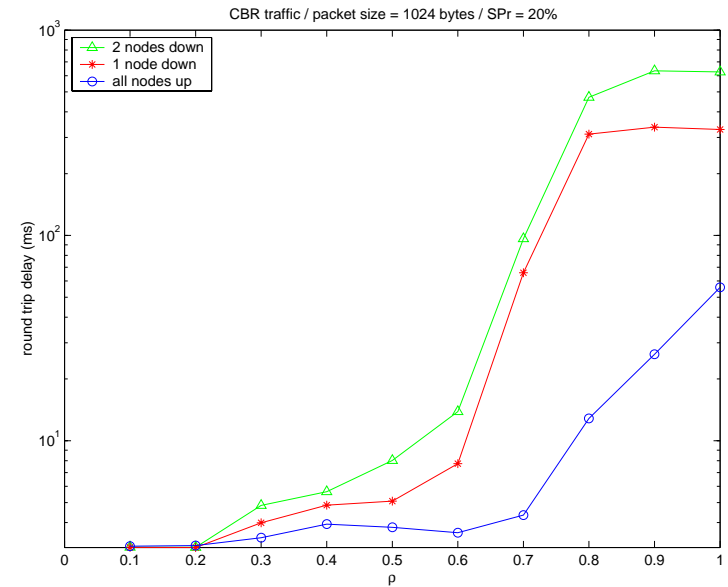
# Packet Losses during Route Reparation



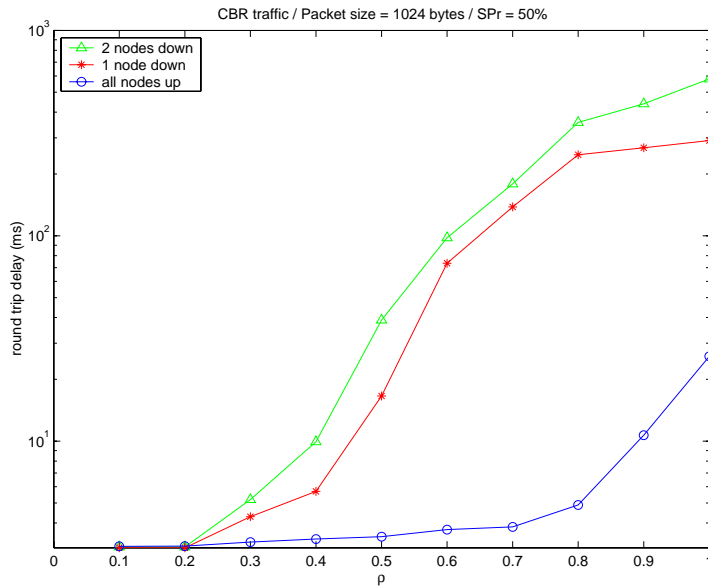
# Round-Trip Delay vs. Offered Load



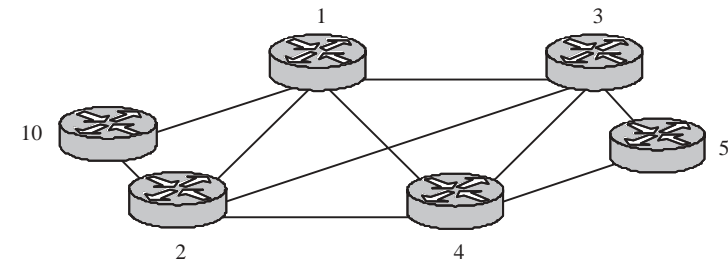
SP ratio = 0.01



SP ratio = 0.20

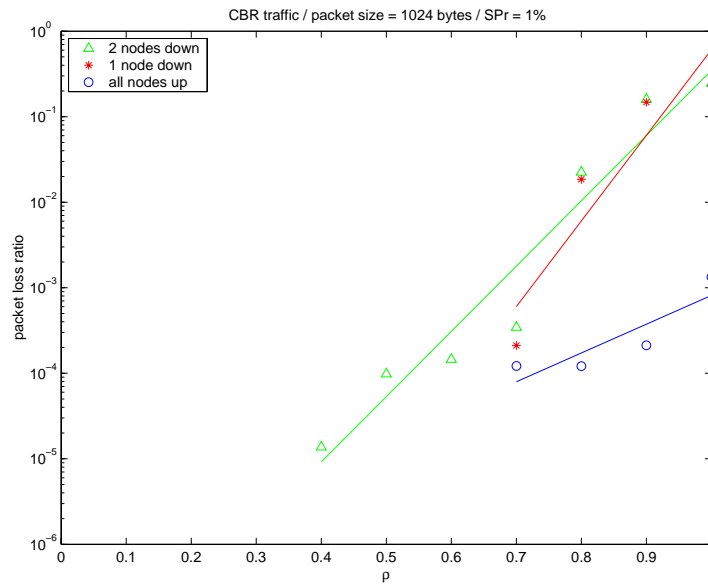


SP ratio = 0.50

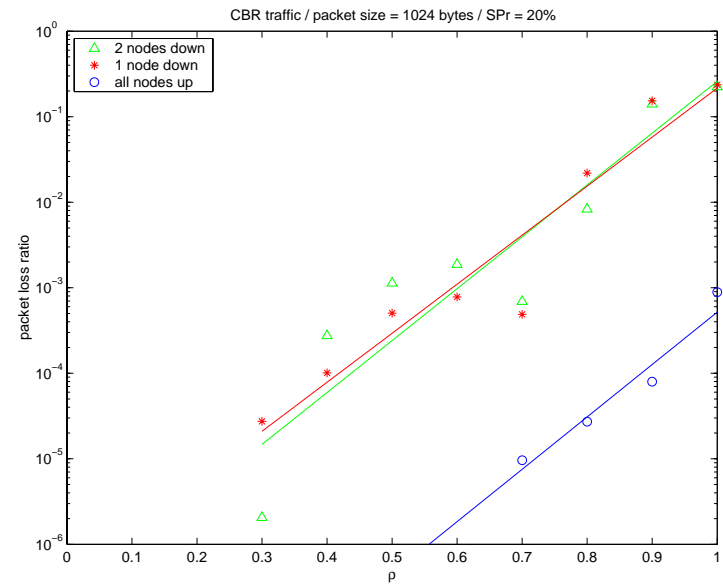


Topology

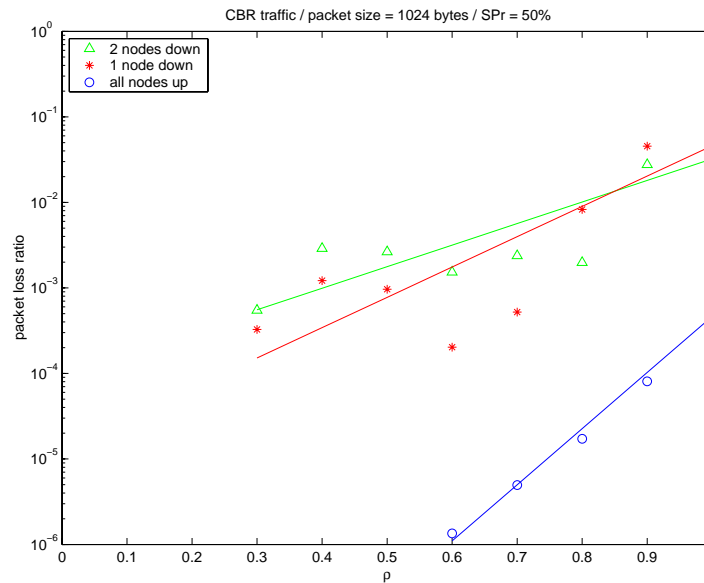
# Packet Loss Ratio vs. Offered Load



SP ratio = 0.01

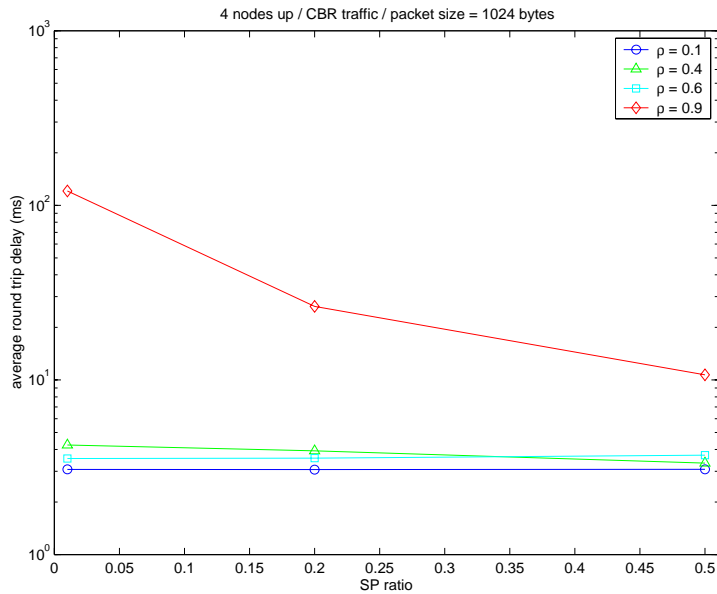


SP ratio = 0.20

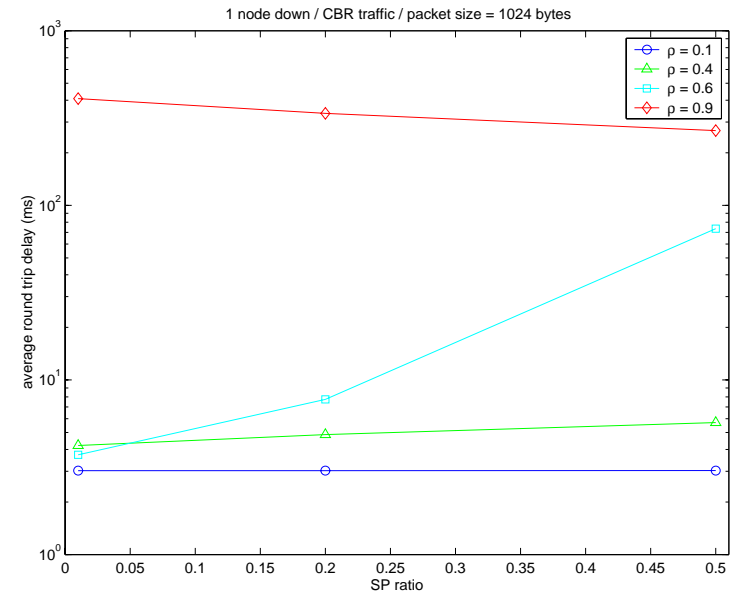


SP ratio = 0.50

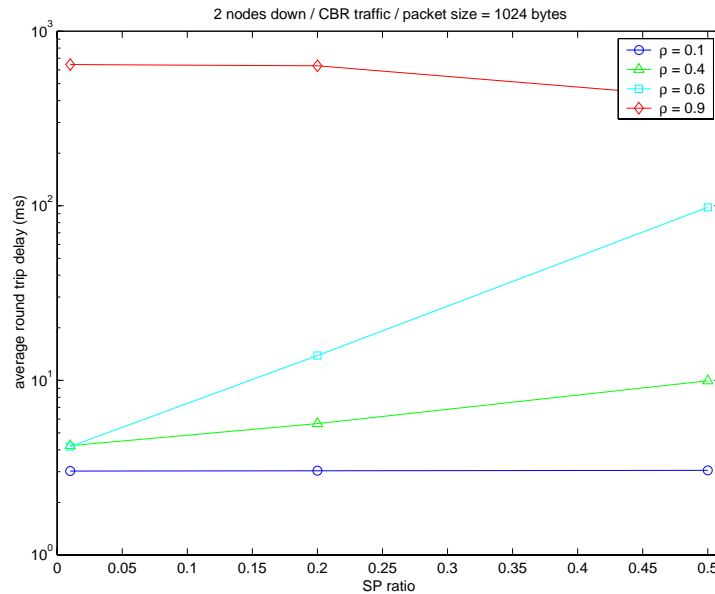
# Round-Trip Delay vs. SP Ratio



no failures



1 core node down



2 core nodes down

# Agenda

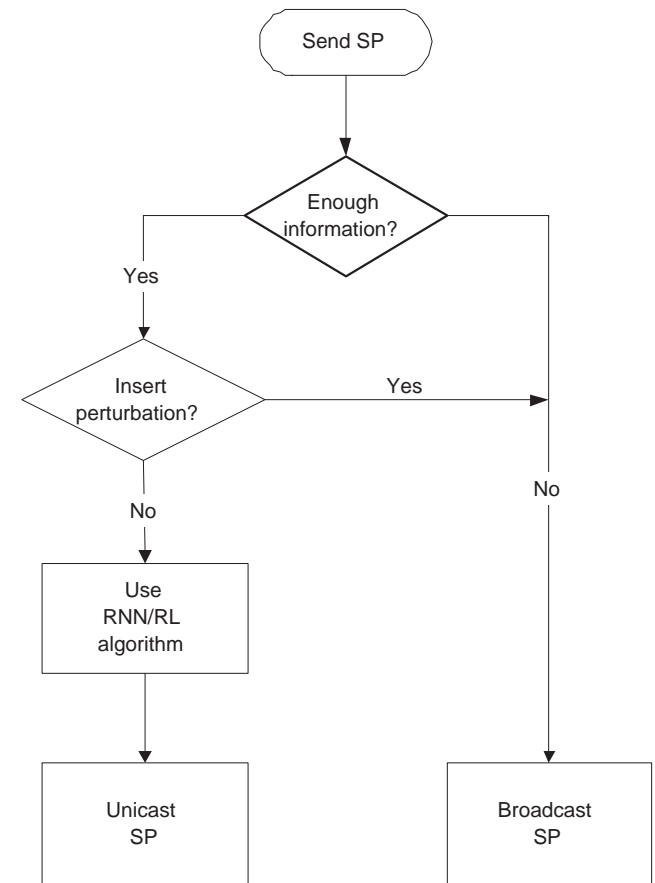
- Introduction to cognitive packet networks (CPN)
- Protocol implementation
- Performance measurements
- **Ad hoc cognitive packet networks (AHCPN)**
- Simulation results
- Summary and conclusions

# Wireless Ad Hoc Networks

- Unreliable system
  - Mobility
  - Wireless connections
  - Shared access (hidden node problem, exposed terminal problem)
  - Nodes with limited resources (battery lifetime)
- Previous proposals flood the network to find destinations (DSR, AODV, ZRP, etc.)

# Ad Hoc CPN

- Layer-2/3 address resolution
  - Listen to transmissions
  - Assign neighbors a time-to-live limit
- SP may use broadcasts
  - Packets are tagged with a packet id
  - Whenever regular RNN/RL is not possible
  - For route maintenance (*broadcast selection ratio*)
- Buffering and Retransmission
- Path availability* in routing goal of smart packets



# Routing Goal

Assume that an SP takes a path  $(n_1, n_2, \dots, n_d)$ , then at any node  $n_i$ :

$$G_i = P_p(n_i, n_d)D(n_i, n_d) + [1 - P_p(n_i, n_d)](T_o + G_i)$$

where

- $G_i =$  routing goal ( $R = G^{-1}$ )
- $P_p(n_i, n_d) =$  path availability from  $n_i$  to  $n_d$  (probability to find a path available for routing)
- $D(n_i, n_d) =$  round-trip delay from  $n_i$  to  $n_d$
- $T_o =$  timeout constant (penalty)

# Routing Goal

$$P_p(n_i, n_d) = \prod_{j=i}^{d-1} P_n(n_{j+1}) P_l(n_j, n_{j+1})$$

$$P_n(n_i) = \frac{B_i}{B_m}$$

- $B_i$  = Battery lifetime at  $n_i$
- $B_m$  = Maximum battery lifetime

$$P_l(n_i, n_{i+1}) = \frac{SNR_m}{SNR(n_i, n_{i+1})}$$

- $SNR(n_i, n_{i+1})$  = signal-to-noise ratio on link  $(n_i, n_{i+1})$
- $SNR_m$  = Receiver SNR sensibility ( $SNR(n_i, n_{i+1}) \geq SNR_m$ )

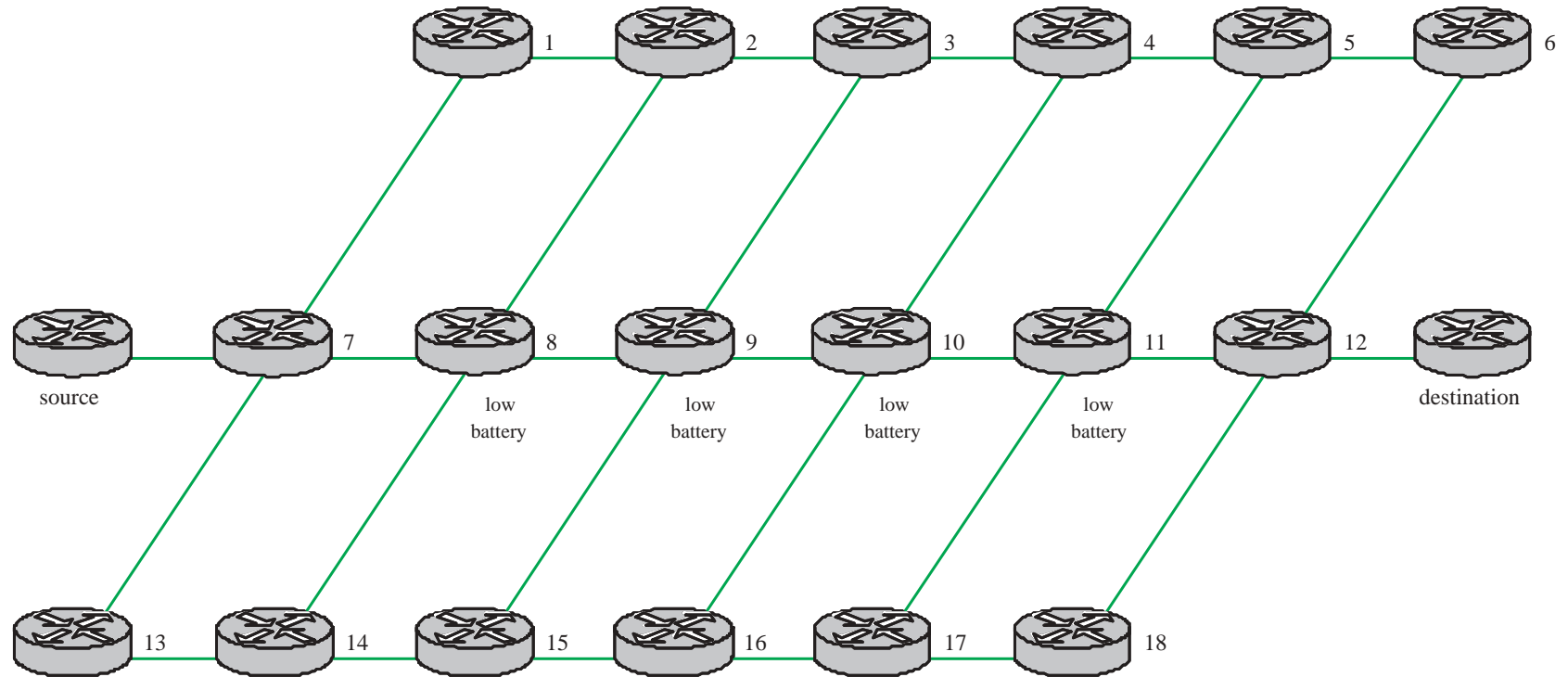
# Agenda

- Introduction to cognitive packet networks (CPN)
- Protocol implementation
- Performance measurements
- Ad hoc cognitive packet networks (AHCPN)
- **Simulation results**
- Summary and conclusions

# Simulation

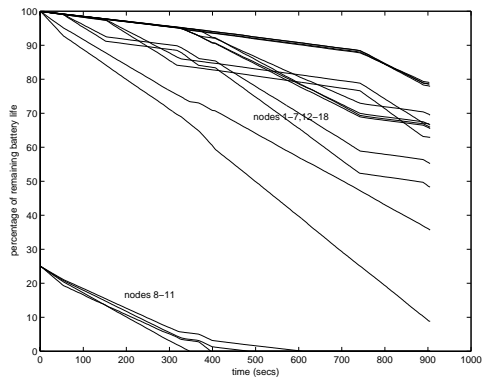
- Experiments on real wireless testbed are hard to conduct
- Simulation model integrated into NS-2
  - The VINT Project (UCB, LBL, USC/ISI, Xerox PARC)
  - C++/oTCL
  - CMU Monarch's wireless extensions

# Energy Awareness - Topology

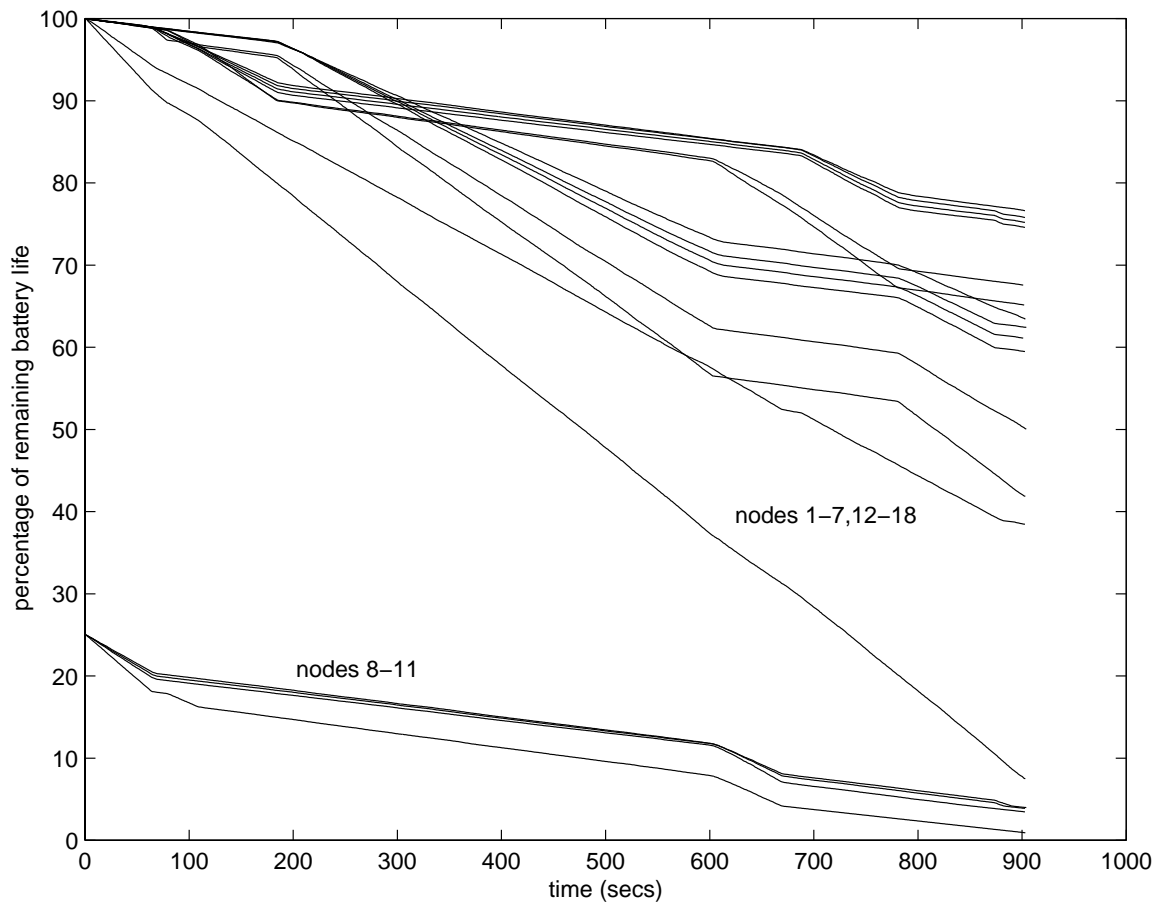


$SP_r = 0.20$ ,  $T_o = 1$ , No retransmission, Battery consumption enabled (high), CBR traffic, packet size = 1000 bytes.

# Battery Lifetime vs. Simulated Time



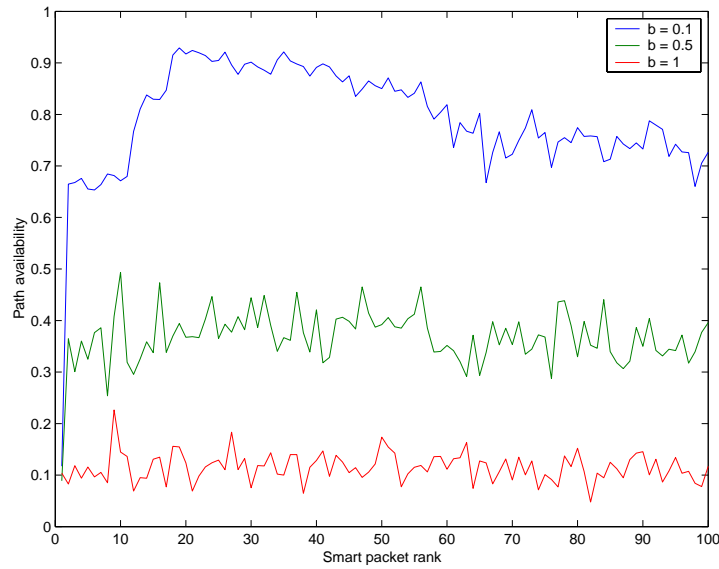
$G = f(D)$



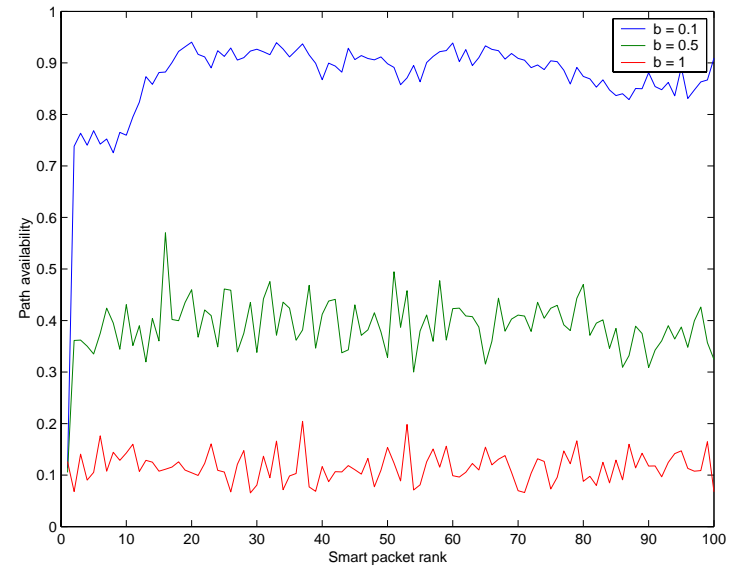
$G = f(D, E)$

$rate = 1pkt/sec, B = 0.01$

# Path Availability vs. SP Rank

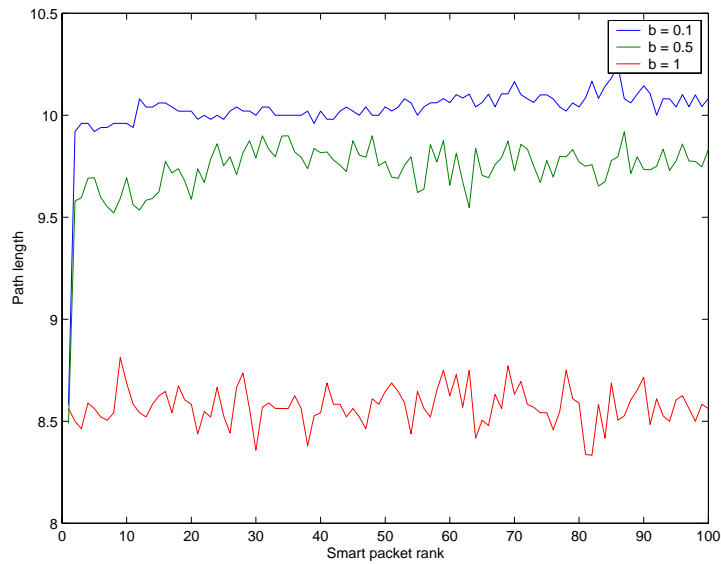


Smart packet rate = 1 pkt/sec

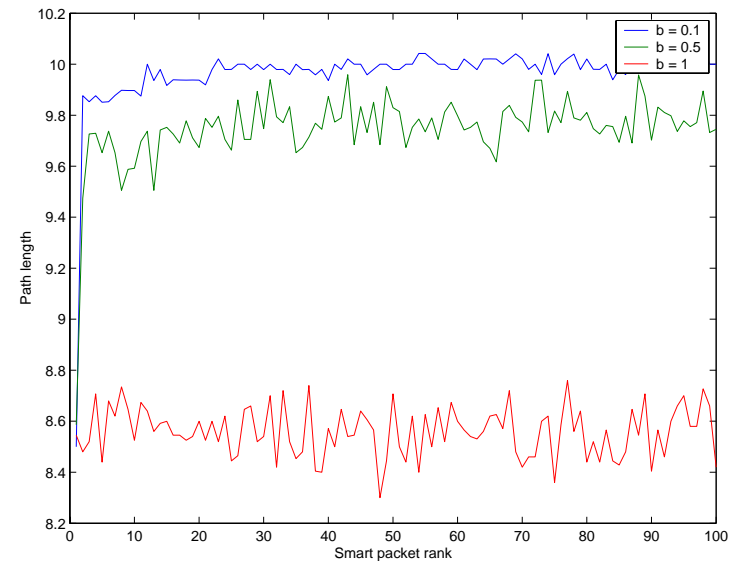


Smart packet rate = 10 pkt/sec

# Average Path Length vs. SP Rank

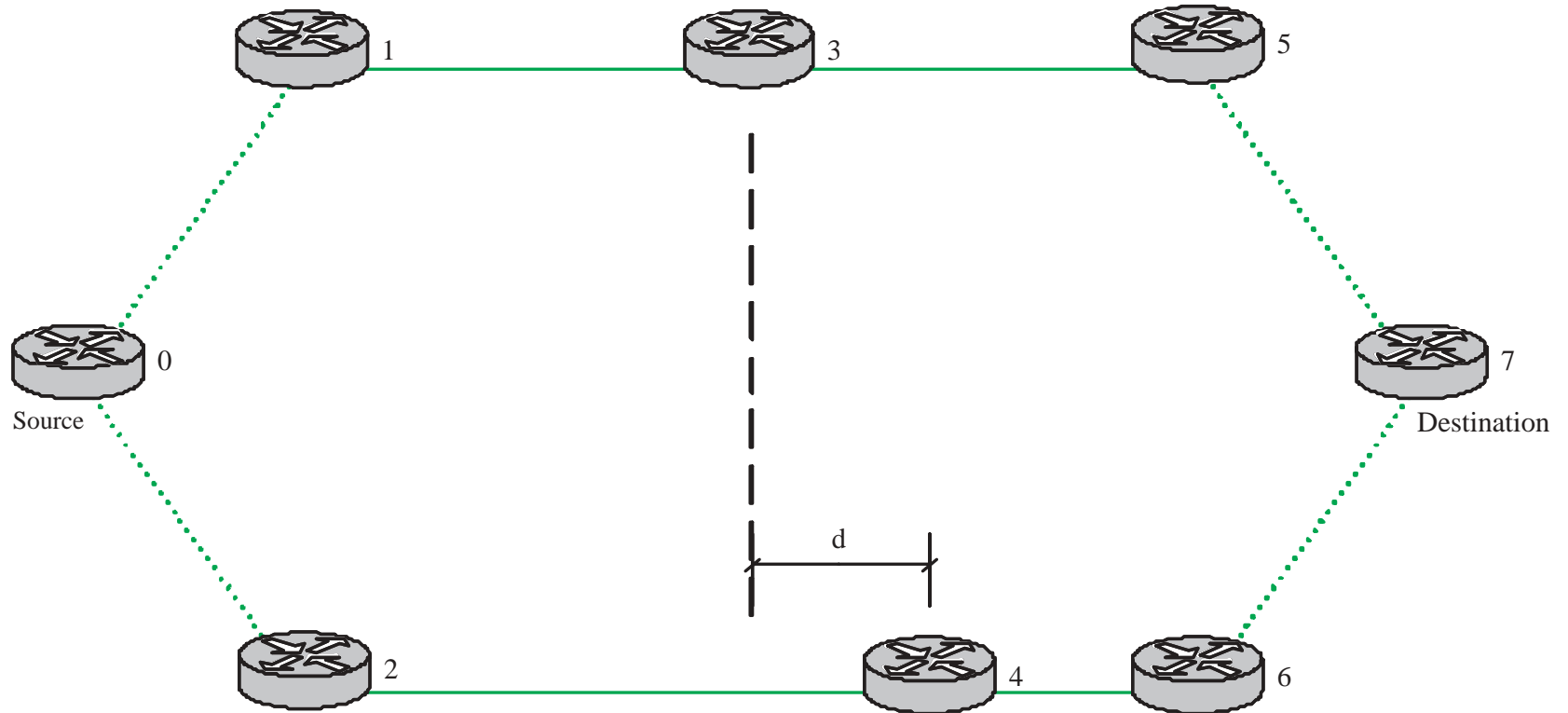


Smart packet rate = 1 pkt/sec



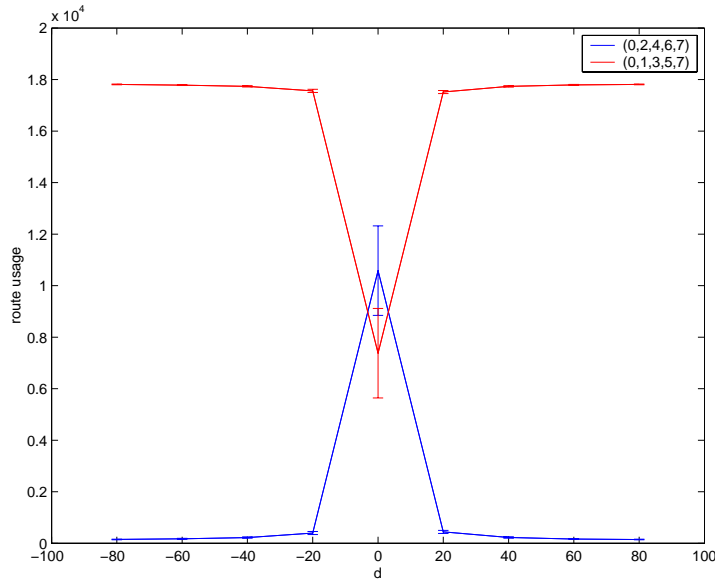
Smart packet rate = 10 pkt/sec

# Link-Q Awareness - Topology

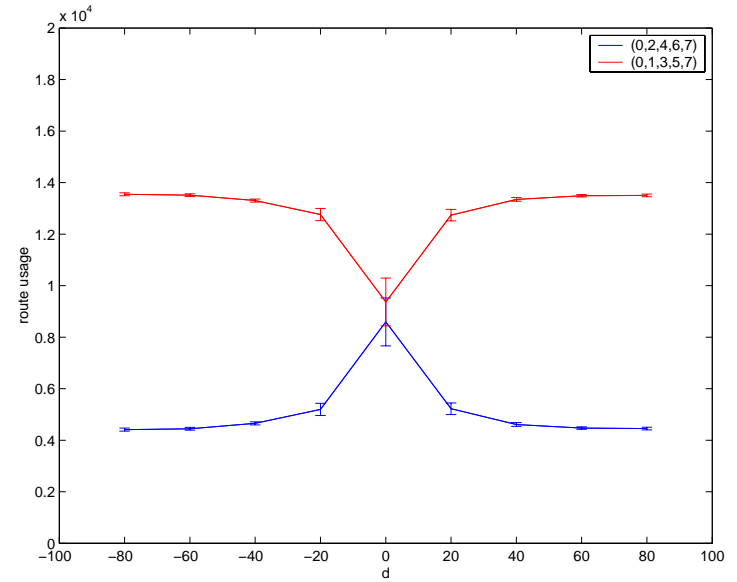


$$SPr = 0.20, T_o = 0.1$$

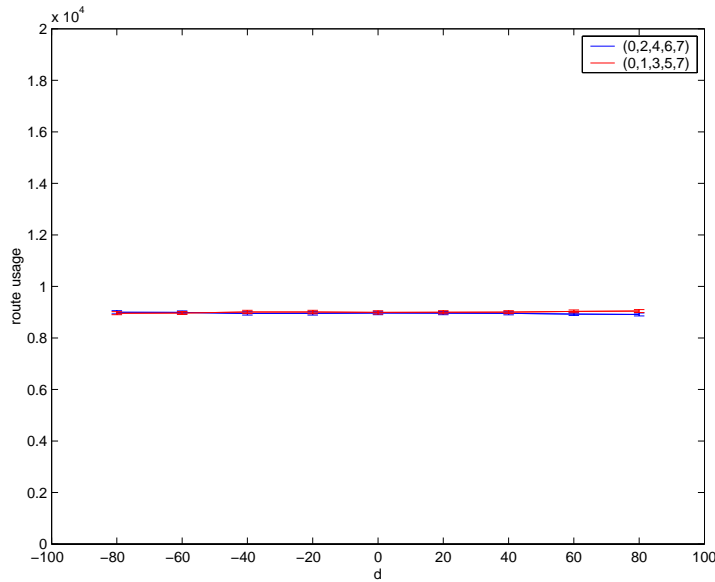
# Path Usage vs. Offset Node 4 (d)



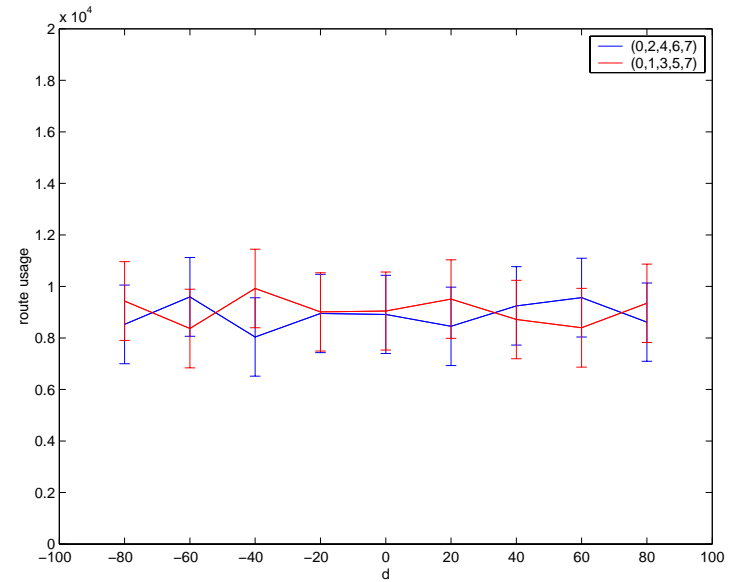
B = 0.01, G = f(D, LQ)



B = 0.50, G = f(D, LQ)



B = 1.00, G = f(D, LQ)



B = 0.01, G = f(D)

# Summary and Conclusions

- Completed design of CPN algorithm for wired networks and wireless, ad hoc networks
  - Route discovery relies on learning (no exchange of routing tables)
  - Suitable for heterogeneous networks (best-effort QoS)
  - Routing criteria was defined as a function of round-trip delay (CPN) and path availability (AHCPN) with a single time reference
  - Introduced randomness and broadcasts in the routing of SP
- Defined and implemented CPN and AHCPN protocols
- Developed a simulation model of AHCPN for NS-2
- Measurements and simulations indicated the great capability for adaptability of CPN and AHCPN
  - In CPN, network performance depends on SP ratio (acceptable values 0.10–0.20)
  - In AHCPN, small SP ratios are preferable. In addition, adaptation level depends on broadcast selection ratio.
  - The introduction of energy awareness and link-quality awareness leads to the establishment of robust paths