# Simulation of multihop, ad hoc wireless networks

DIMACS Workshop on Modeling and Simulation

Oct 23-25, 1997

Princeton, NJ

Mario Gerla

Computer Science Dept, UCLA

Los Angeles,CA 90024

Research supported by DARPA (WAMIS, GloMo programs); by Intel and by Northrop

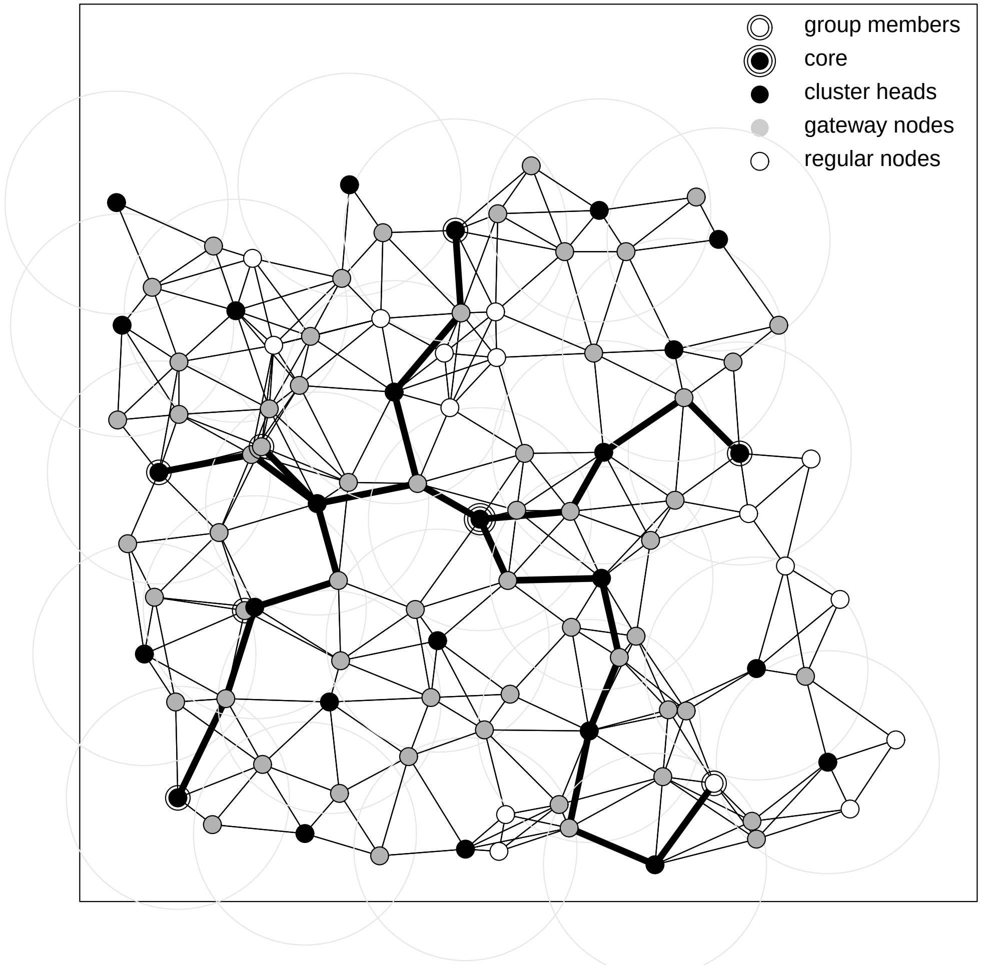
# Why Simulation of Wireless Networks

- Wireless network design requires careful tradeoffs between many variables
- Strong interdependence of protocol designs at different layers
- Analytic models are inadequate
  (either computationally impractical or oversimplified)
- Simulation can be a pathway to implementation

## Challenges of wireless network simulation

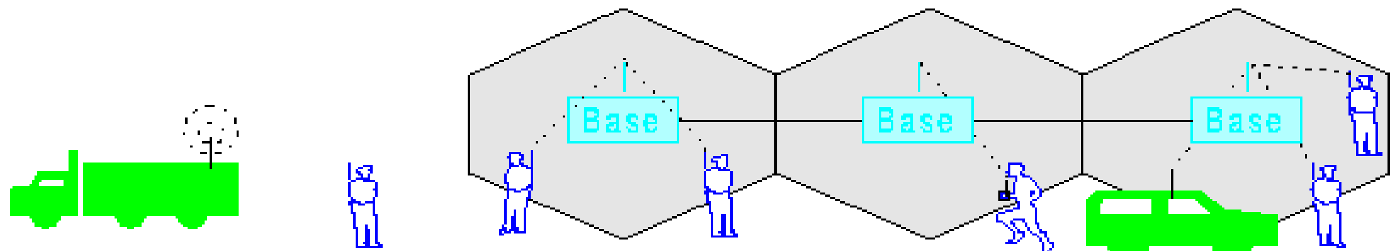
- mobility
- impredictable radio channel quality
- multi-access channel interference
- multimedia traffic support
- built-in protocol adaptivity to network changes
- multihopping
- scaling to large populations

# Example: Wireless Multicast

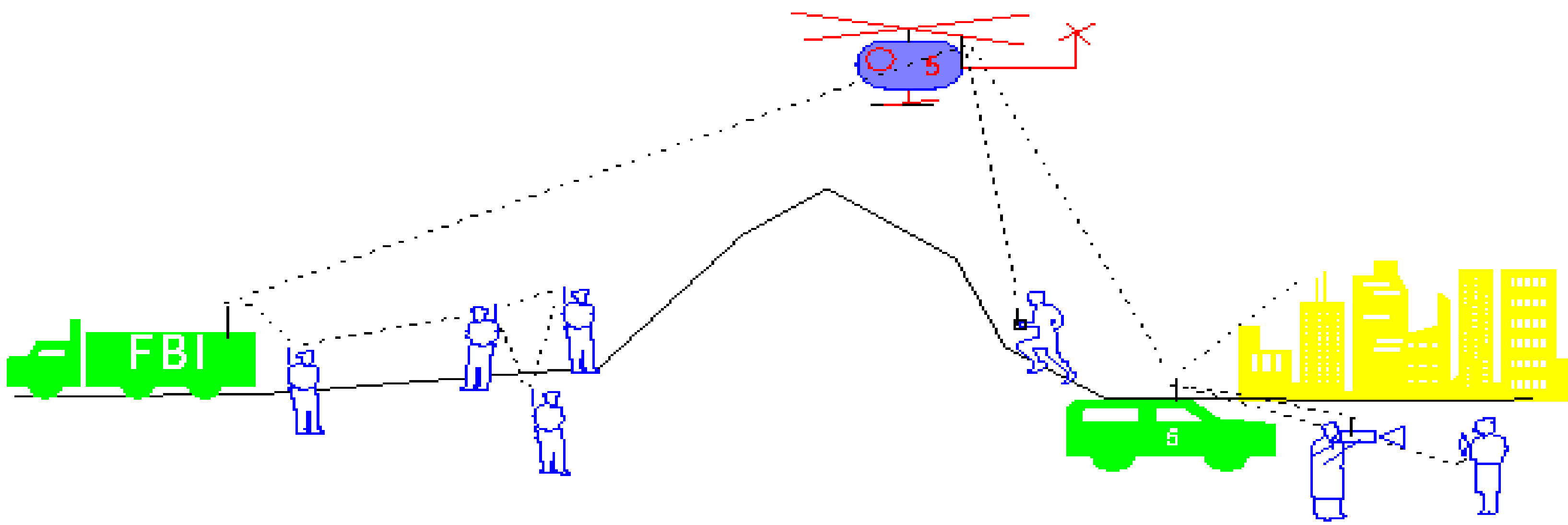


# Cellular vs Multihop

- Standard Base-Station Cellular Networks




- Instant Infrastructure, Multihop wireless Networks



## Target Applications

Military:

battlefield, sensors

Law Enforcement:

border patrol, crowd control

Civilian:

disaster (earthquake, fire, flood) recovery

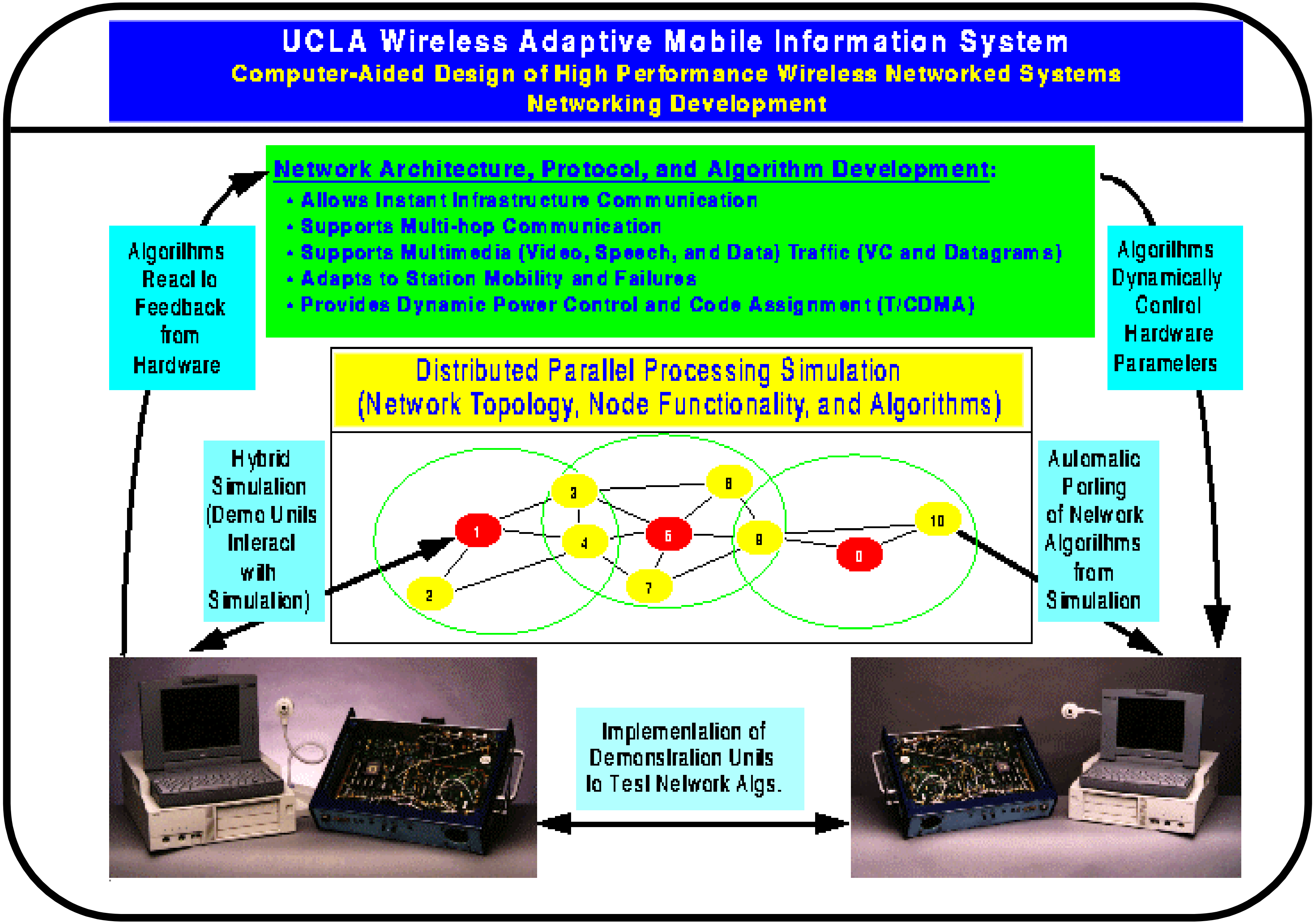
ad hoc networking

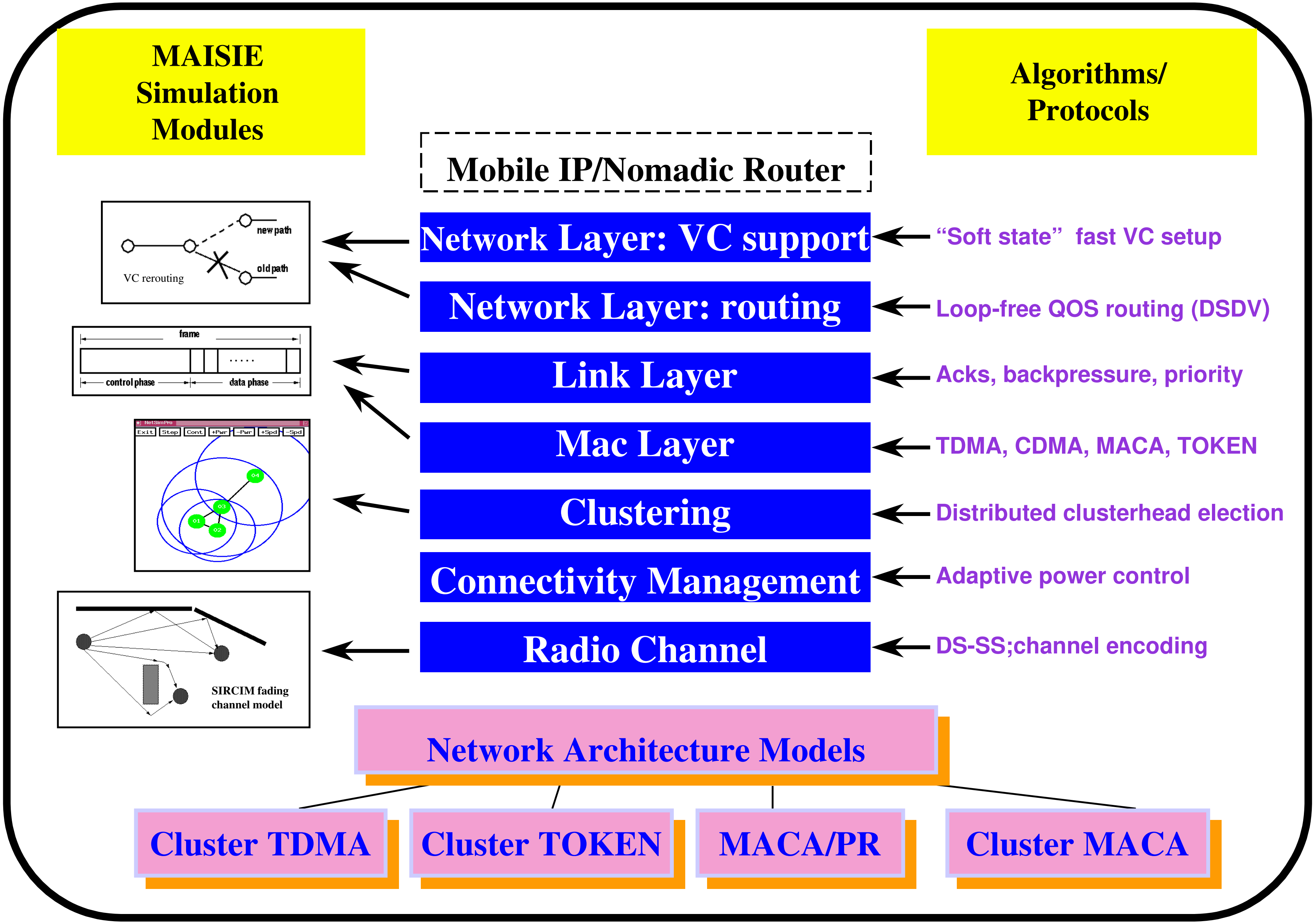
# Requirements

mobile users
rapid development ⇒ Mobile infrastructure
reconfigurable topology

low power
spatial reuse ⇒ Multihop
obstacles

emergency operations
real time interaction ⇒ Multimedia
man in the loop


UCLA Wireless Adaptive Mobile Information System
Computer-Aided Design of High Performance Wireless Networked Systems
Networking Development
Network Architecture, Protocol, and Algorithm Development:
• Allows Instant Infrastructure Communication
• Supports Multi-hop Communication
• Supports Multimedia (Video, Speech, and Data) Traffic (VC and Datagrams)
• Adapts to Station Mobility and Failures
• Provides Dynamic Power Control and Code Assignment (T/CDMA)
Algorithms React to Feedback from Hardware
Algorithms Dynamically Control Hardware Parameters
Distributed Parallel Processing Simulation
(Network Topology, Node Functionality, and Algorithms)
Hybrid Simulation (Demo Units Interact with Simulation)
Automatic Porting of Network Algorithms from Simulation
Implementation of Demonstration Units to Test Network Algs.


MAISIE Simulation Modules
Algorithms/ Protocols
Mobile IP/Nomadic Router
new path
old path
VC rerouting
frame
control phase
data phase
SIRCIM fading channel model
Network Layer: VC support
Network Layer: routing
Link Layer
Mac Layer
Clustering
Connectivity Management
Radio Channel
"Soft state" fast VC setup
Loop-free QOS routing (DSDV)
Acks, backpressure, priority
TDMA, CDMA, MACA, TOKEN
Distributed clusterhead election
Adaptive power control
DS-SS;channel encoding
Network Architecture Models
Cluster TDMA
Cluster TOKEN
MACA/PR
Cluster MACA

# Cluster Network Architecture for WAMIS

- **Concept**

  create a cluster based TDM infrastructure which:

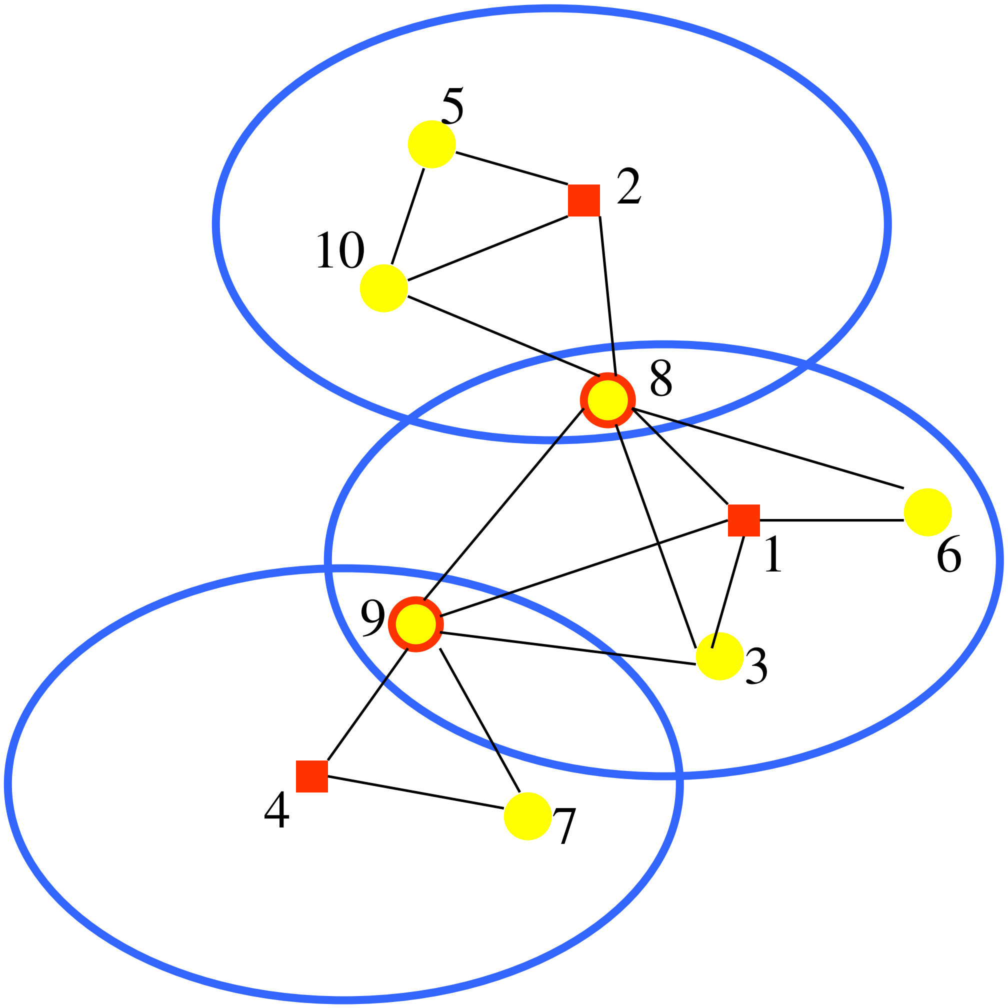
  (a) enables guaranteed bandwidth for voice/video

  (b) can support instant deployment and mobility

- **Approach**
  - distributed clustering algorithm
  - time division slotting within each cluster
  - slot reservation for real time traffic
  - virtual circuits for real time traffic; datagrams for data traffic
  - code separation across clusters
  - slot synchronization

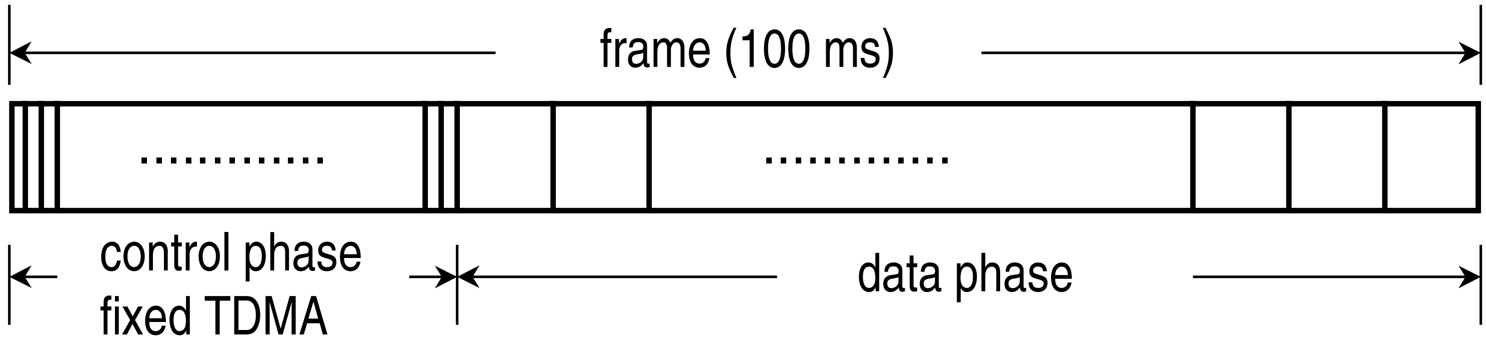
**combines cellular radio and traditional packet radio features**

## Clustering example



# Channel Access

Within each cluster: time-slotted frame




**Control phase**:

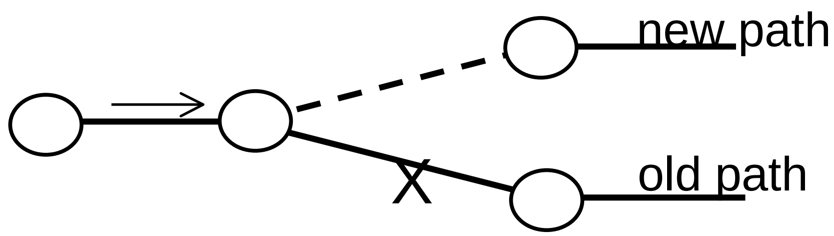
- clustering algorithm
- routing update
- power measurement and control
- code and slot assignment
- VC setup
- acknowledgments

**Data phase**:

- voice/video (PRMA)
- data (Random Access)

# VC Reconfiguration in Mobile Environment

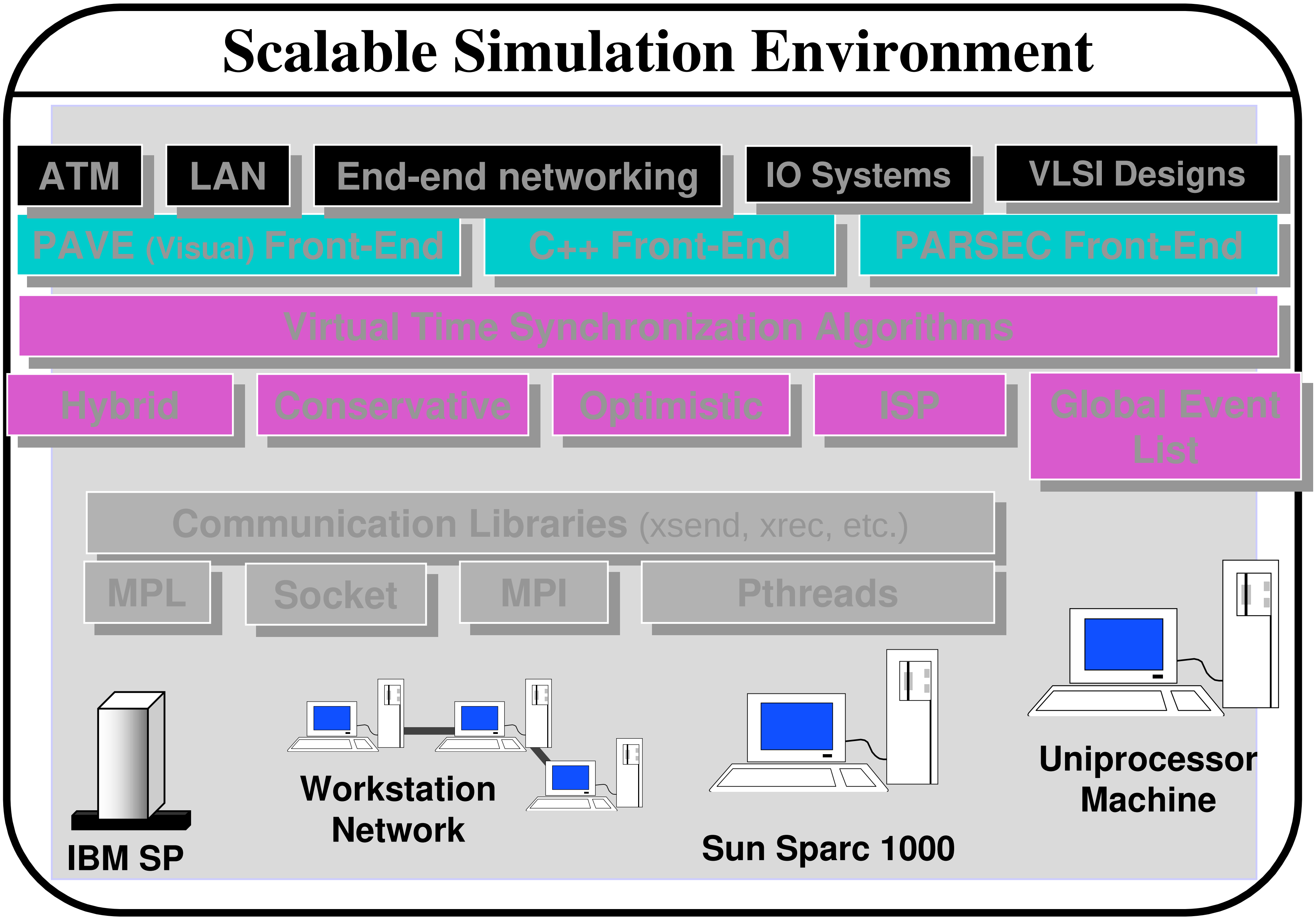
- Conventional VC setup does not work(path breaks up too  frequently)
- Proposed approach: Fast Reservations, like in PRMA ( Packet Reservation Multiple Access)




- First packet reserves the slot(s) along the path
- When path changes, first packet competes again for slots on new path (voice/video rate reduced by low priority pkt drop)
- If no path, packet is dropped
- Reservation released if slot is unused

# Maisie/PARSEC

- C-based general purpose DES language
- Graphical front-end for model design & execution
- Domain-specific libraries:
  - queueing networks
  - parallel architectures and I/O systems
  - wireless networks, ...
- Support for diverse sequential (Solaris, AIX, Linux, IRIX, …), and parallel machines(Sparcstations, SGI Origin 2000, IBM SP2, Myrinet NOWs, …) and protocols (conservative, optimistic, hybrid)
- Migration of models to implementations



Scalable Simulation Environment
ATM
LAN
End-end networking
IO Systems
VLSI Designs
PAVE (Visual) Front-End
C++ Front-End
PARSEC Front-End
Virtual Time Synchronization Algorithms
Hybrid
Conservative
Optimistic
ISP
Global Event List
Communication Libraries (xsend, xrec, etc.)
MPL
Socket
MPI
Pthreads
IBM SP
Workstation Network
Sun Sparc 1000
Uniprocessor Machine

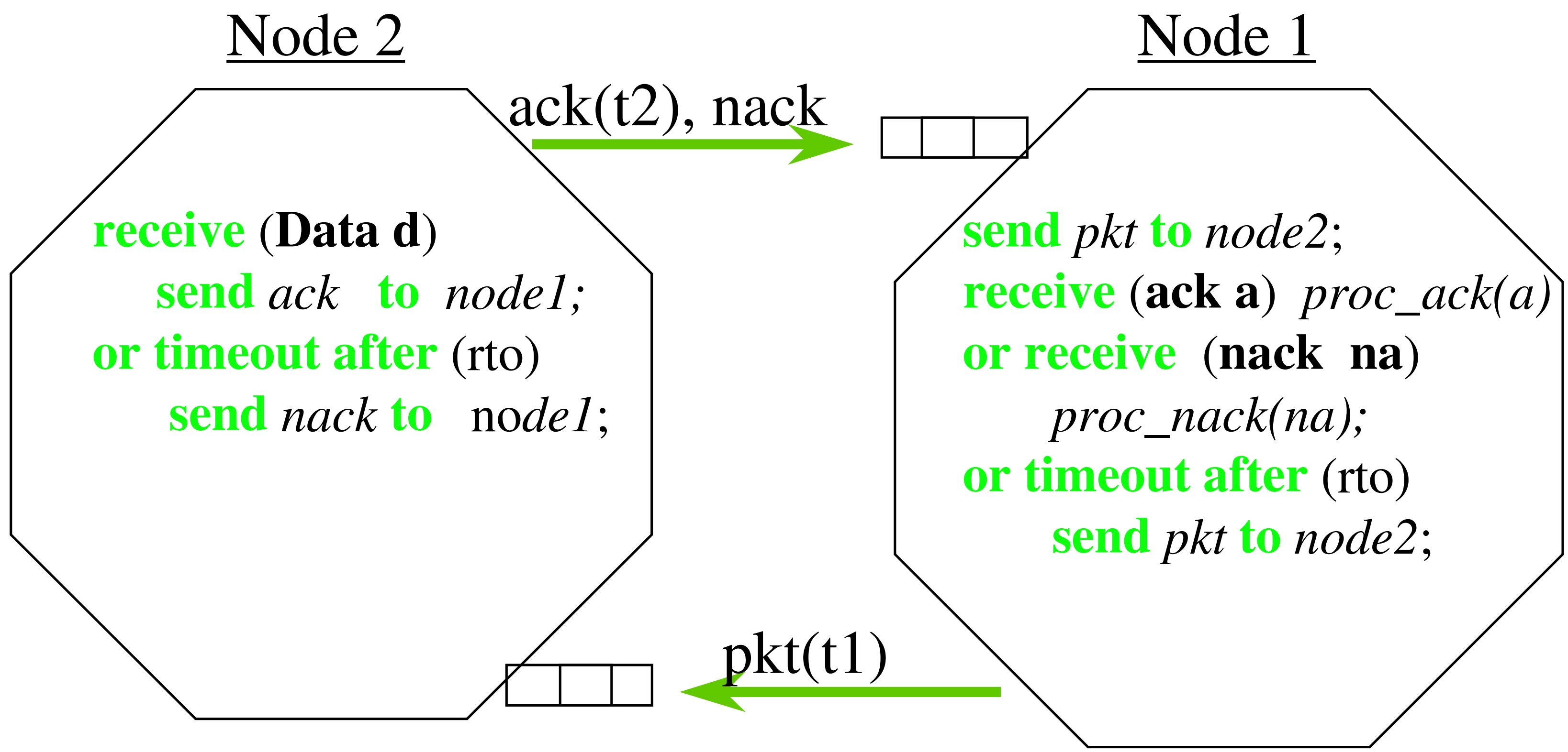
# Maisie/PARSEC

## Message-based, process-interaction approach

- LP: programmed as an *actor* or *entity*
  - all activities of an *entity* initiated on message receipts
  - An LP cannot directly modify the state of another LP (only via messages)
- event: represented by message communications
  - e(t,p,a): schedule message *m(e)* at time *t* for LP *p*
- on receiving *m(e)*, *p* executes actions *a*
  - change its state (local variables)
  - schedule events(messages) at *time >=t*
  - unschedule previously scheduled events (e.g. *timeout*) ;

    only *future* events may be unscheduled

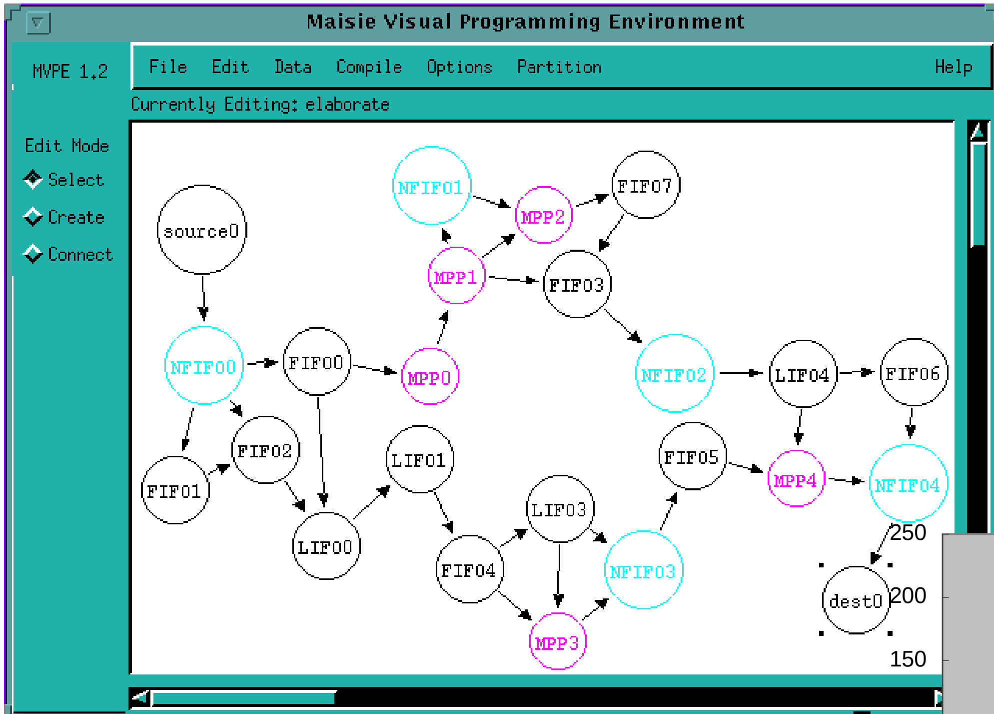
# PARSEC Code Sample



# Visual Simulation Environment

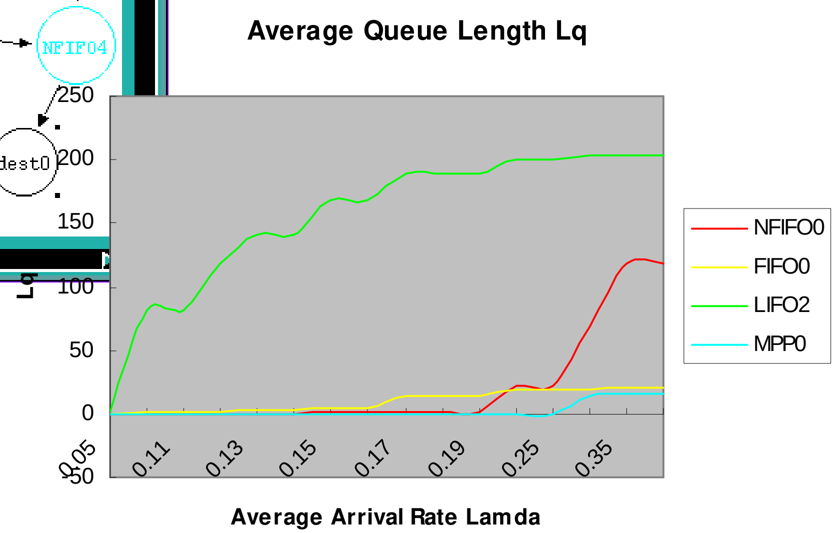
- PARSEC Visual Environment (PAVE)
  - general purpose visual simulation environment.
  - built-in support for scalability and parallelism
  - customizable
- Detailed Entity Design
  - helps with parallel s imulation inserts
- Hierarchical Modeling
  - replicated entity structures
- Partitioning
  - user assisted
  - automatic

# Model Design & Execution




## MVPE Features

- point and click model building
- hierarchical construction
- rapid reconfiguration
- visual model partitioning
- interactive control of parallel sim

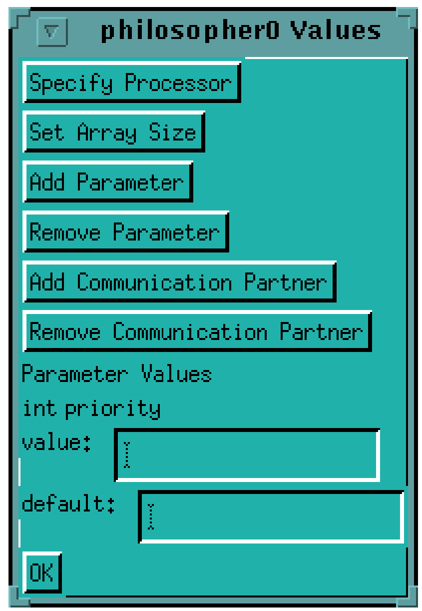



## Queue Model Outputs

- system throughput
- mean queue length, and more

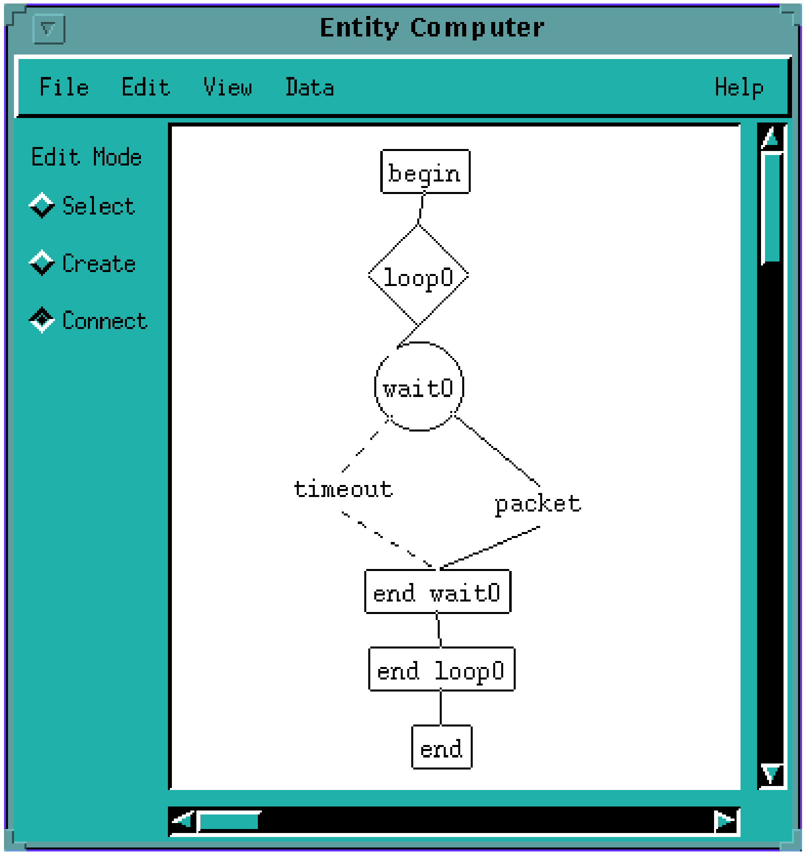
# Interface Specification

- Example of message definition
- and specification of communication partners

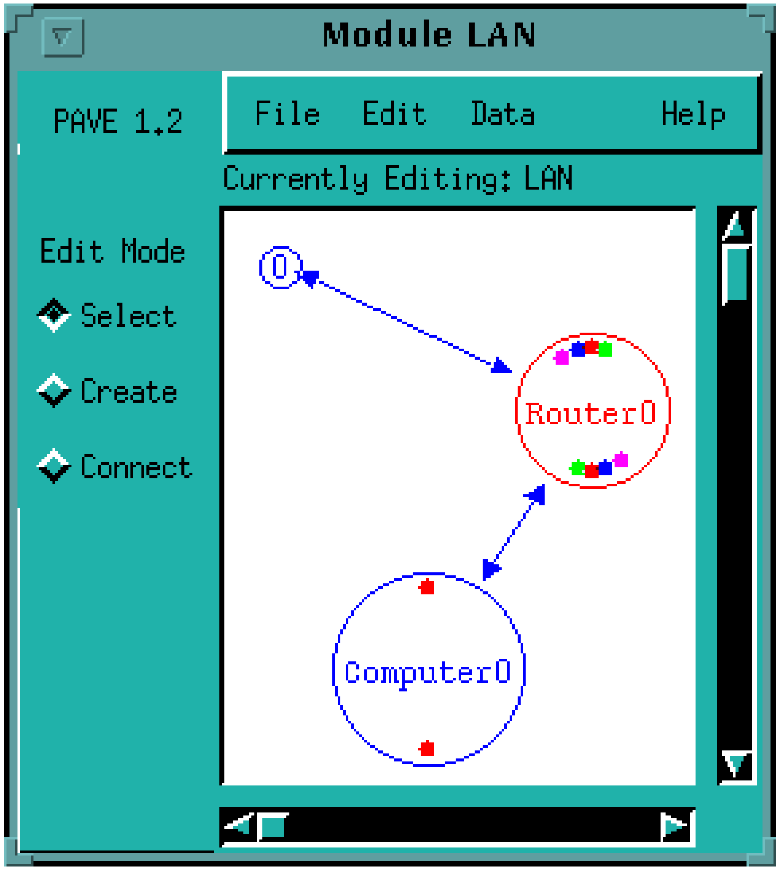



# Internal Processing

- Flow chart notation includes
  - squares for code blocks
  - circles for receiving messages
  - arrows for sending messages

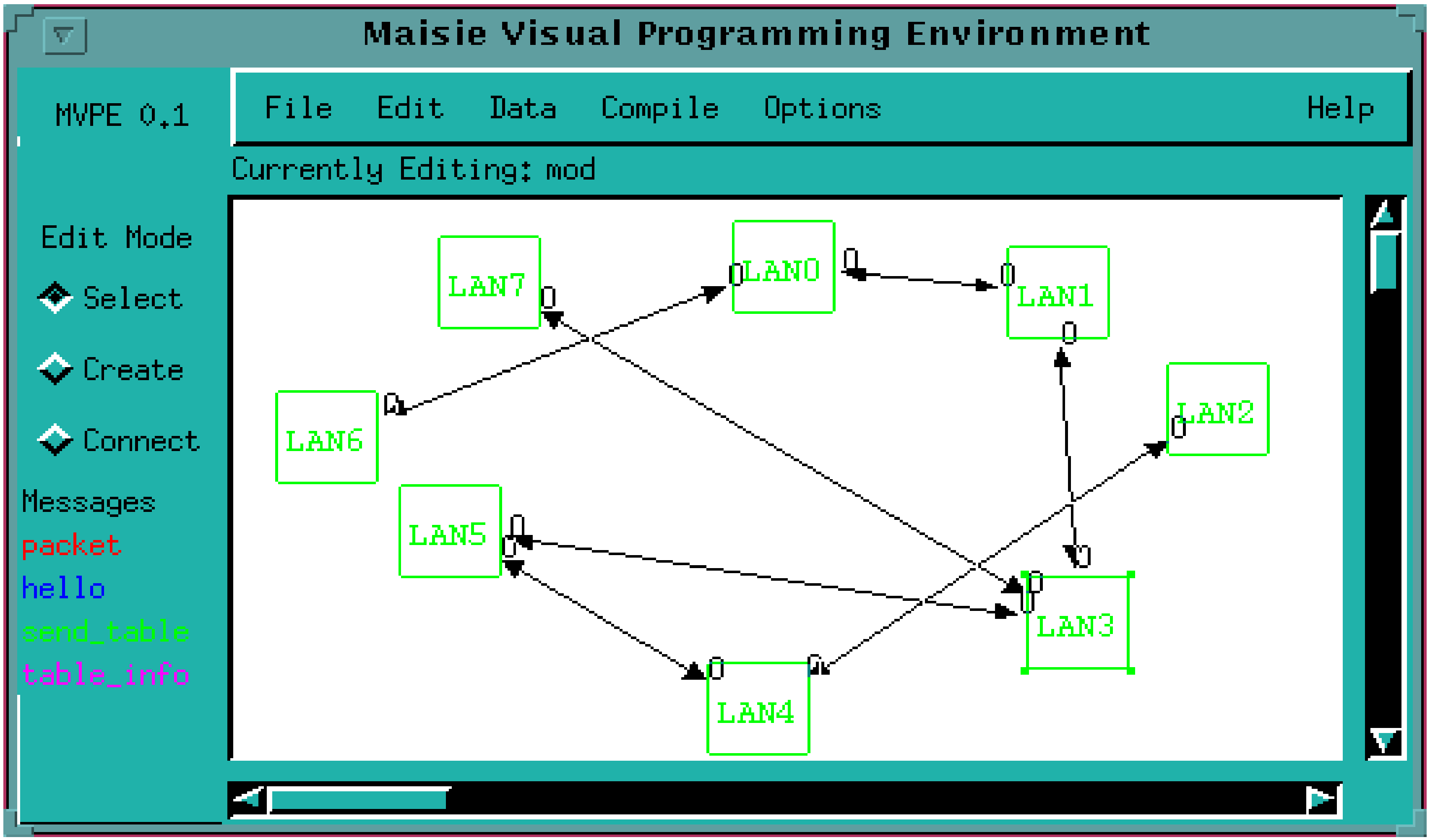


# Module Building




- The Computer is connected only to the Router
- The Router communicates out of the module through the I/O port.

# Topology Specification



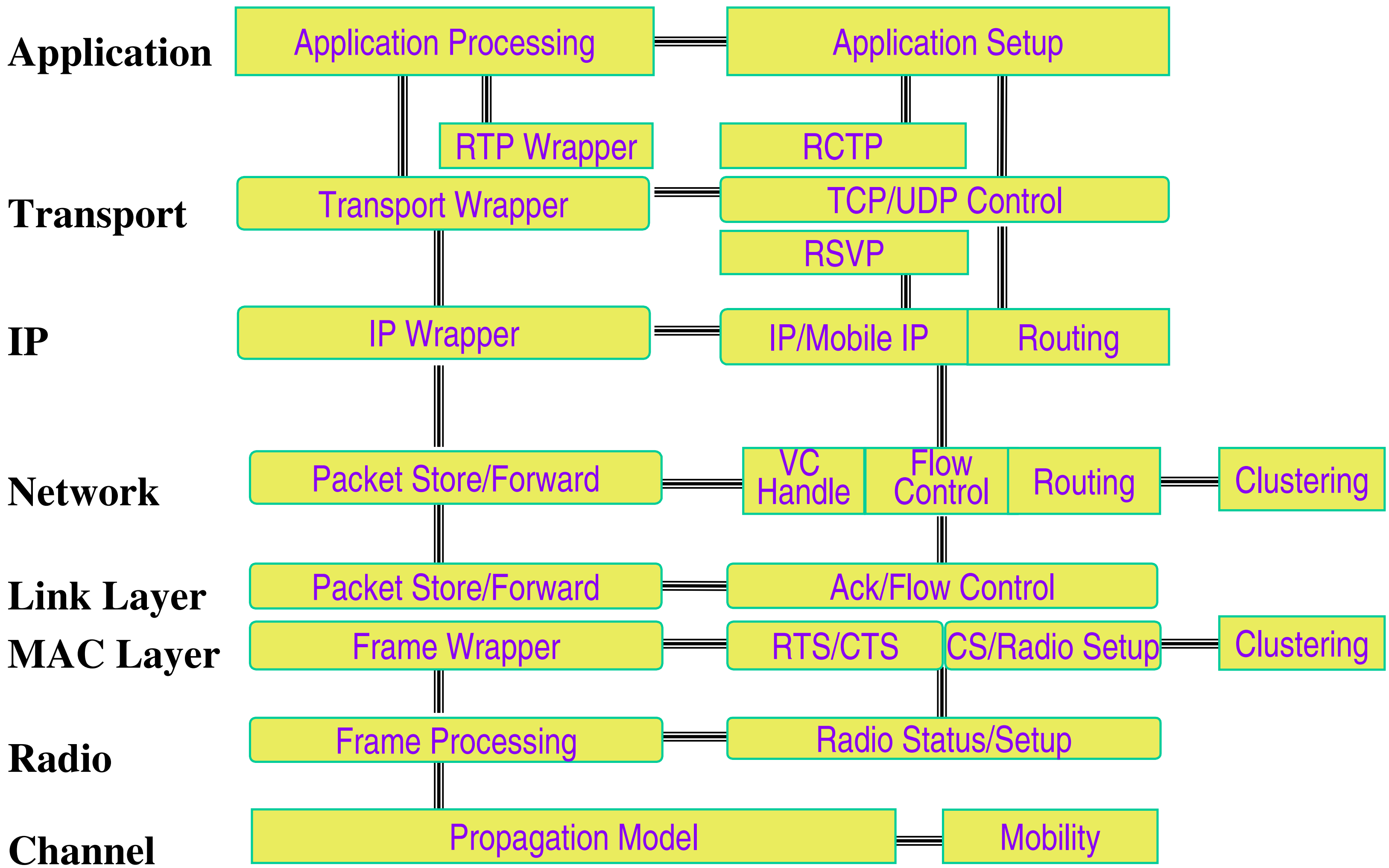
# GloMoSim Platform

- Modular, layered stack
- Standard APIs
- Library of modules
- layer-specific performance metrics

Objectives:

- common platform for testing and comparison of GloMo protocols
- scaling of testbeds

# Glomosim Simulation Layers



# Application: Comparison of Multihop Protocols

Synchronous ⟷ Asynchronous

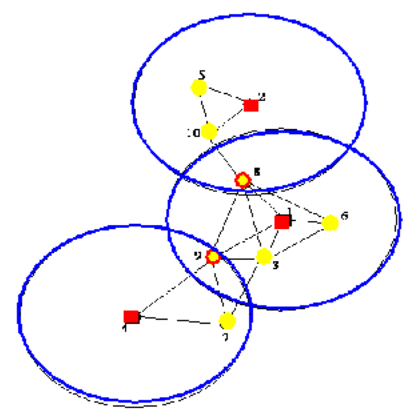
| Cluster TDMA | Cluster Token | MACA/PR | PRNET |
|---|---|---|---|
| Global synchronization | Cluster synchronization | Dialog synchronization | No synchronization |

# DARPA PRNET (1970's)

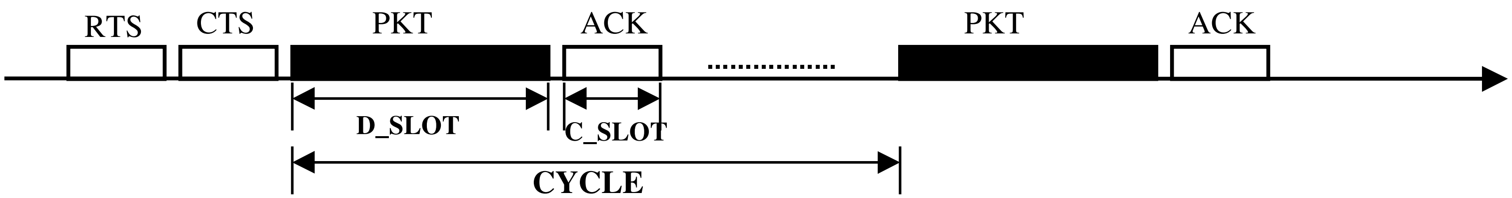
- CSMA (hidden terminal)
- Implicit ACK (echo ACK)
- Flow control: pacing (re)transmission protocol
- Routing: B-F routing scheme
- Duct routing (for voice traffic)
  - multiple paths carry multiple copies of voice packet to improve reliability
  - Carrier sense will limit the fan-out

- Limitations of PRNET
  - no bandwidth reservation; no access control (for voice)
  - “hidden terminal” problem
- Enter Cluster TDMA (1994)
  - different codes in each cluster
  - TDMA type MAC access in each cluster
  - QoS routing; bdw reservation; access control
  - Fast VC setup (soft state)

*Cluster TDMA*



- Problems of cluster TDMA: cost and complexity
  - global slot synchronization
  - multiple codes
  - initialization
- Cluster Token: no slot synchronization
- MACA/PR (Multiple Access Collision Avoidance/Packet Reservation)
  - even simpler than Cluster Token
  - no clustering; single code; easy initialization
  - RTS/CTS dialog ( to prevent “hidden terminal” problems)
  - Packet Reservations ( to support real time traffic)



# Overall Performance Comparison

| | PRNET | MACA/ PR | Cluster Token | Cluster TDMA |
|---|---|---|---|---|
| avg VC thrput (pkt/sec) | 4.6 | 9.69 | 9.88 | 9.97 |
| Total thrput (pkts/sec) | 14.54 | 34.45 | 95.34 | 107.70 |
| end-to-end delay (ms) | 73.05 | 66.50 | 298.80 | 302.13 |
| avg pkt loss per vc | 64.30% | 3.07% | 1.18% | 0.02% |

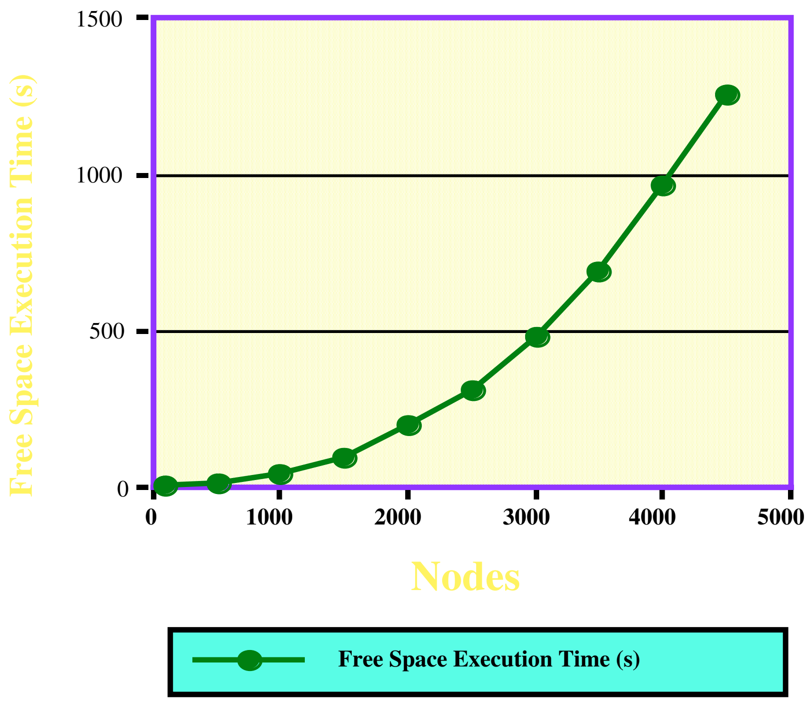
- PRNET
  - No bandwidth reservation
  - No acceptance control
  - In heavy load: duct routing generates excessive number of "requests for alternate routes" ( congestion)
- MACA/PR
  - total VC throughput limited by lack of cluster/code separation
- Cluster TOKEN and TDMA
  - high end to end delay due to token/TDMA latency

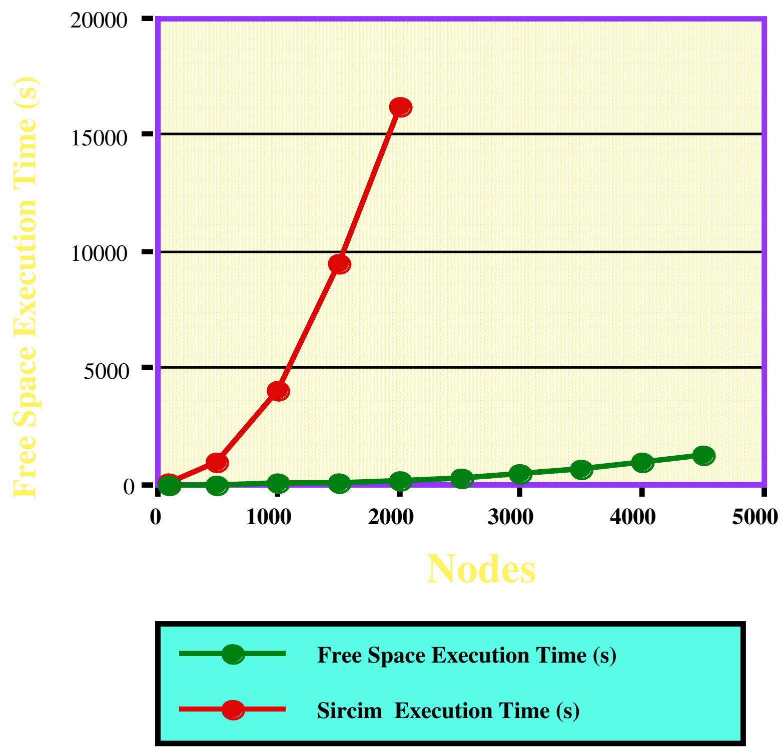
# Addressing Scalability

- Hierarchical modeling
- Parallel execution
- Hybrid modeling

# PARSEC Scalability

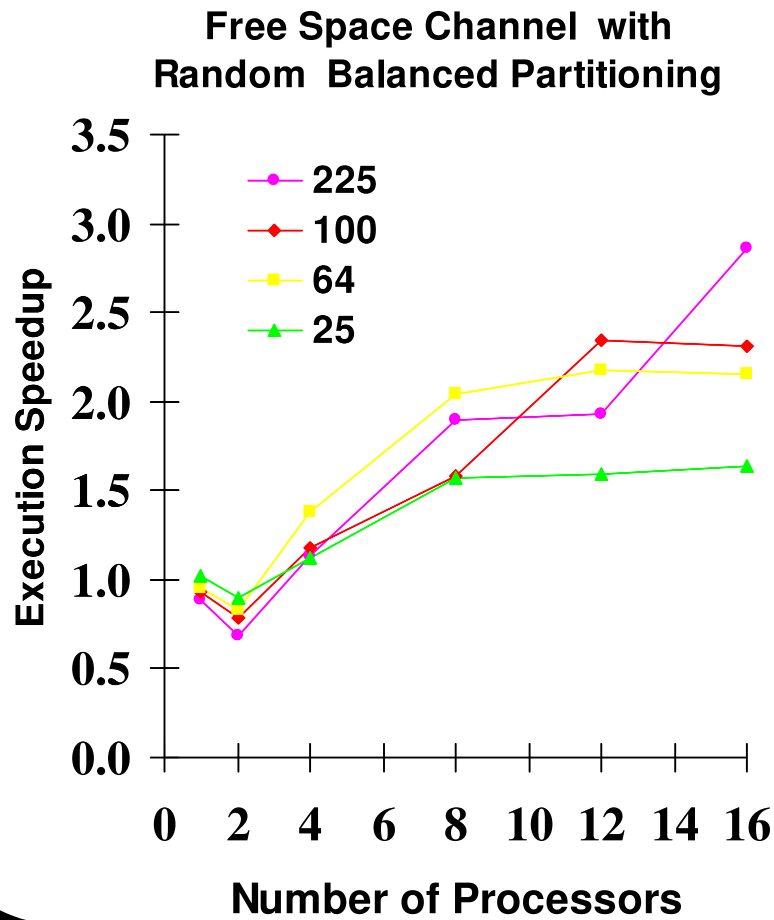
- CSMA model with free space channel models
- Scaling demonstrated to **4500** nodes
- Fading channel model 30-50 times slower

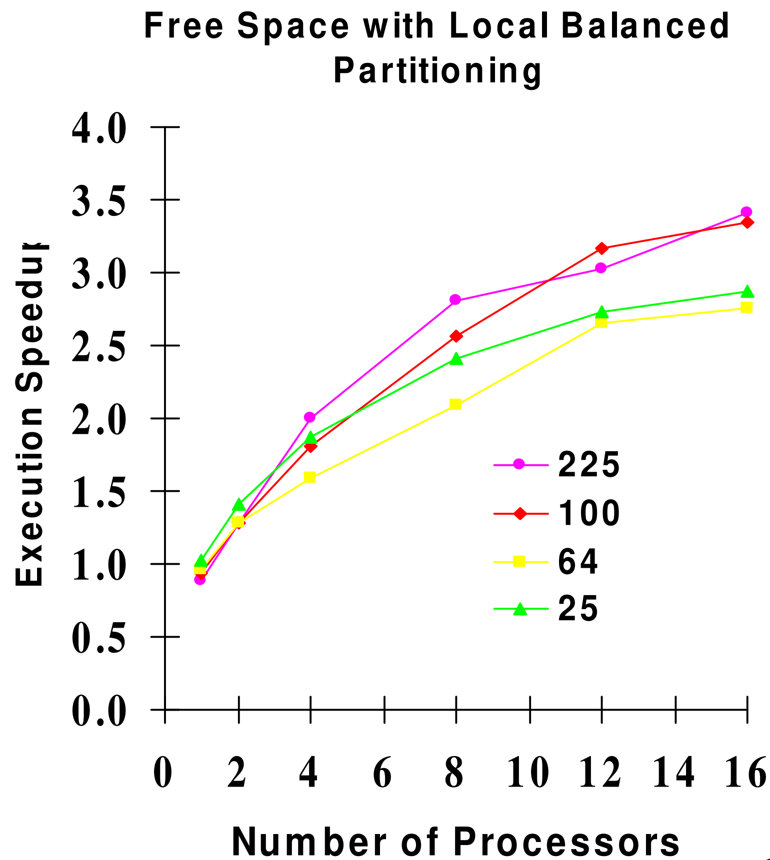


# Parallel Simulation Experiments

- Network sizes: 25, 64, 100, and 225 nodes
- Simulation Platform:
  - IBM SP2 multicomputer
- Parallel performance as a function of network partitioning

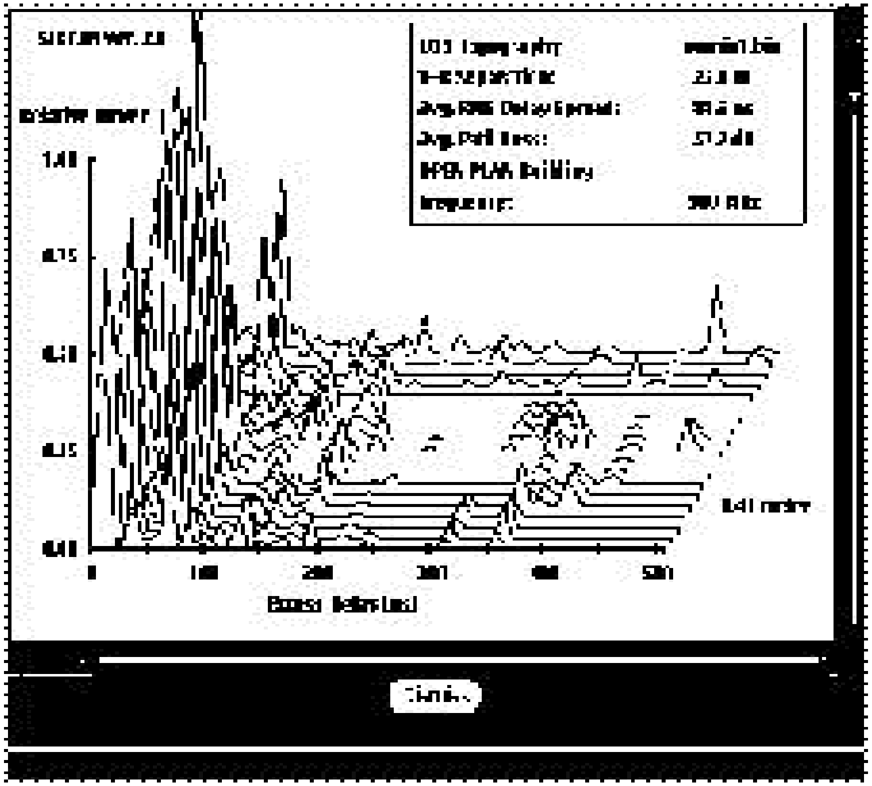


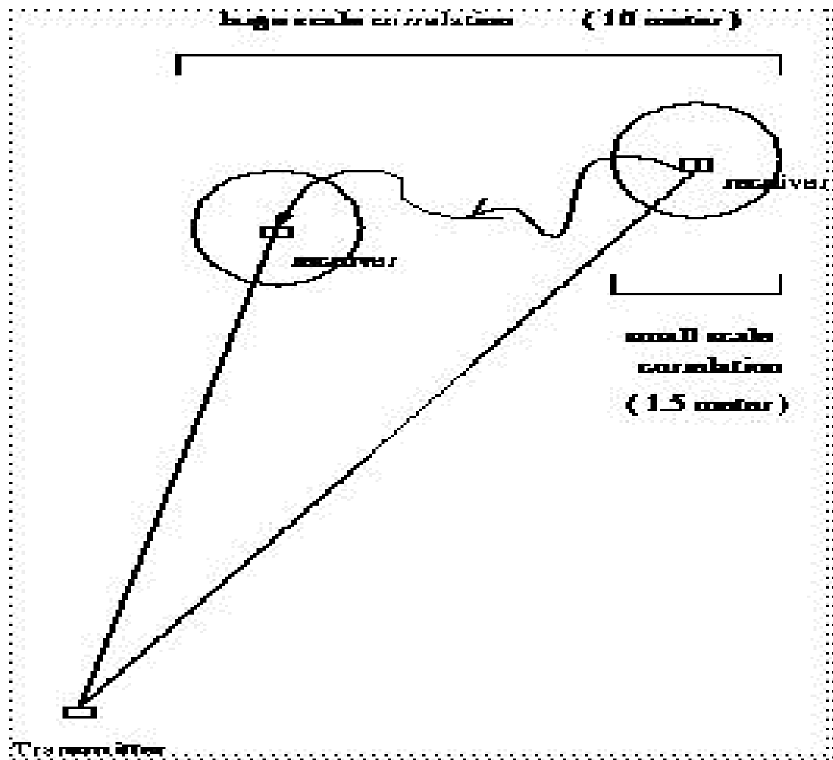



# Simulator: Maisie Channel Model

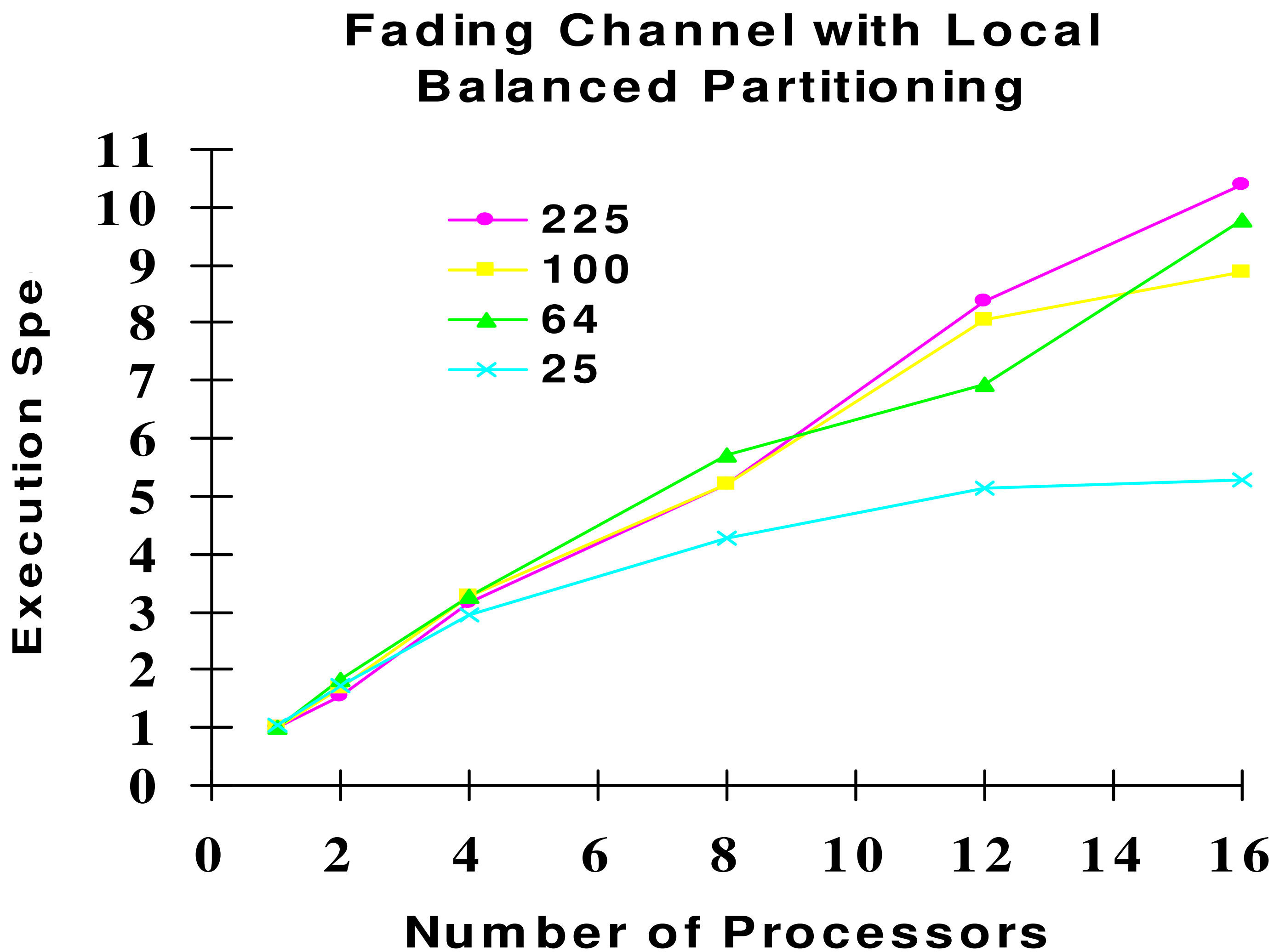
## Channel Fading Model in Maisie Simulatior

***The Simulator utilizes the SIRCIM impulse response parameters to characterize the radio propagation model, i.e.: multipath, shadowing effect, spatial correlation***



# Results


Fading Channel with Local
Balanced Partitioning
225
100
64
25
Execution Spe
Number of Processors
0
1
2
3
4
5
6
7
8
9
10
11
0
2
4
6
8
10
12
14
16

# Work in Progress

- expand GoMoSim library (mobility models, antenna/radio/channel models, IP, RSVP, TCP etc)
- hybrid simulation
- measurement postprocessing routines
- testbed implementation (Linux and NT)
- wireless multicast (reliable, QoS, internet/wireless)
- QoS routing