Performance Evaluation of Multiple Criteria Routing Algorithms in Large PNNI ATM Network

Phongsak Prasithsangaree

Master's Thesis Defense

May 22, 2000

Presentation Flow

- What is PNNI?
- Problem statement
- Our solution
- Implementation
- Performance metrics
- Test scenarios
- Performance evaluation
- Conclusions

What is PNNI ?

- Private Network to Network Interface
  – A protocol for ATM networks
- PNNI is composed of two protocols
  – PNNI Routing Protocol
  – PNNI Signaling Protocol

PNNI Routing Protocol

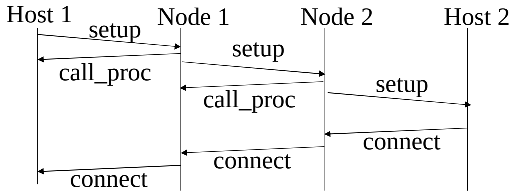
- Hello Protocol
  – Hello Packets are exchanged between neighbor nodes
  – To discover and verify the identity of the neighbor nodes.
- Flooding Mechanism
  – A reliable hop-by-hop propagation of topology information.
  – Topology information, PTSE, and PTSP
  – PTSE is subject to aging and is removed after a pre-defined duration.
  – An updated PTSE is sent when topology information is *significantly* changed.
  – A *significant change* is determined by configuration parameters.

PNNI Routing Protocol (continued)

- PNNI Topology Metrics and Attributes
  - Metric: delay or administrative weight
  - Attribute: bandwidth, CLR, or CDV
- Routing Mechanism
  - Router gets a route request with requirements.
  - Router retrieves topology information from its database.
  - Router finds a possible path according to the requirements.
  - Return the path in the DTL format.

PNNI Signaling Protocol

- A subset of UNI 4.0 signaling standard.
- Call Setup Procedure
- Call Admission Control (CAC)
- Crankback and Alternate Routing



Problem Statement

Problem Statement

- The popular routing algorithm used is Dijkstra’s algorithm, which can find a path based on a single cost.
- Need something better than Dijkstra’s Algorithm
- Multiple QoS Routing
  - A routing algorithm that can find a route with more than one constraint at the same time.
- However...
  - The problem of deciding if there is a path which satisfies more than one additive constraints is *NP-complete.*

Our Solution

Our Solution

- Heuristic Multiple Criteria Routing Algorithms

Secondary Criterion →

Primary Criterion ↓

	Widest	Shortest	Minimum Hop
Widest	○	×	×
Shortest	×	○	×
Minimum Hop	×	×	○

○ Single QoS Routing Algorithm

× Multiple QoS Routing Algorithm

Implementation

Implementation

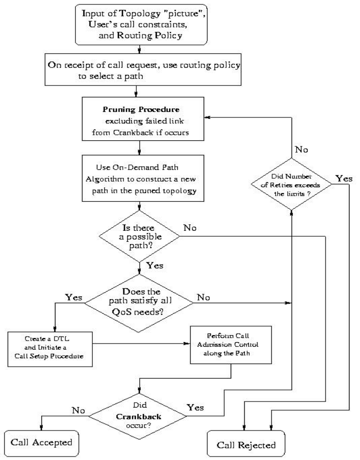
- Generic Call Admission Control (GCAC)
  - Standardized by ATM forum to be used for call admission control (CAC).
  - CAC is vendor-specific.
  - GCAC is used to reduce the routing computational time.
  - It prunes links and nodes that cannot support the call.

Routing Algorithms

- Dijkstra's algorithm has cost and distance as parameters.
- Widest Shortest Algorithm
  – Modified Dijkstra's algorithms to consider two costs and two distances.
- D_widest algorithm
  – modified relaxation method of Dijkstra's algorithm
- Shortest Widest Algorithm (has two routing passes)
  – The first pass used D_widest algorithm and the second pass used the modified Dijkstra's algorithm

Implementation (continued)

- Routing Computation Flow Chart



Performance Metrics

Performance Metrics

- Average Call Failure Rate

$$= \frac{\text{Total number of rejected calls}}{\text{Total number of requested calls}}$$

- Average Call Setup Time

$$= \frac{\text{Total call setup time}}{\text{Total number of successful calls}}$$

Performance Metrics (continued)

- Routing Inaccuracy

$$= \frac{\text{Number of Crankback events}}{\text{Total number of call requests}}$$

- Link Utilization

$$= \frac{\text{Used Link Bandwidth}}{\text{Link Bandwidth Capacity}}$$

Test Scenarios

Test Scenarios

- Edge-Core Networks

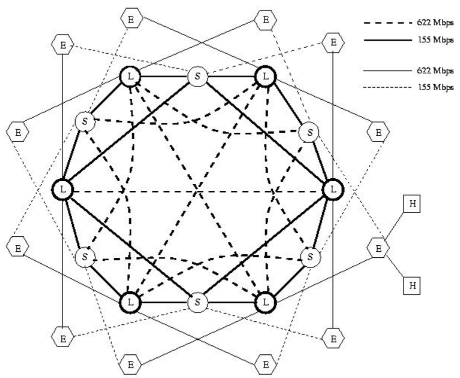
Link	Capacity	Delay (ms)
L node to L node	OC-12	Uniform[25,40]
S node to L node	OC-3 or OC-12	Uniform[10,25]
E node to L node	OC-12	Uniform[5,10]
E node to S node	OC-3	Uniform[5,10]

Table 4.2: Link Metrics for Conventional Edge-core Topologies

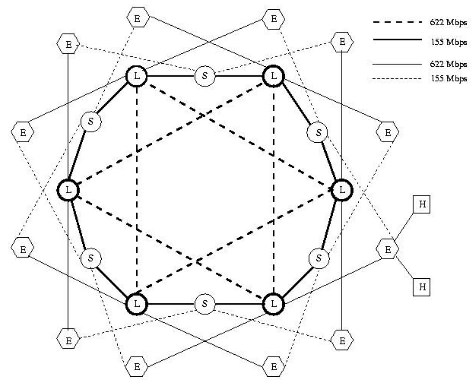
Topology	Nodes	Link Type	Links	Connectivity
Light	12 core 12 edge	S-L L-L E-L E-S	12 6 12 12	1.75
Dense	12 core 12 edge	S-L L-L E-L E-S	24 3 12 12	2.125

Table 4.3: Summary of Edge-Core Topologies

Dense Edge-Core Network



Light Edge-Core Network



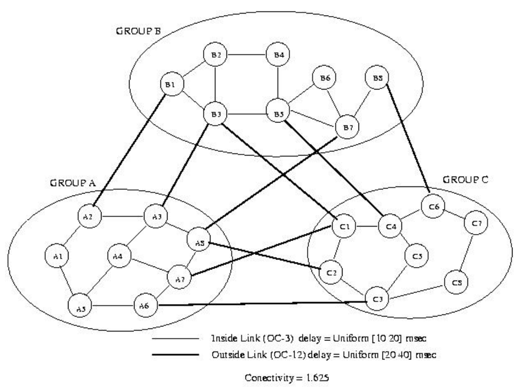
Test Scenarios

- Cluster Networks

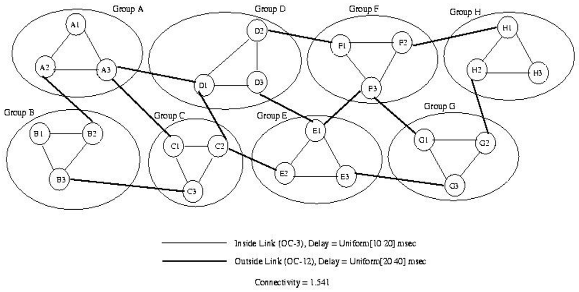
Topology	nodes	Link Type	links	Bandwidth	Delay	Connectivity
3-cluster	24	Outside	9	OC-12	Uniform[20 40]	1.625
		Inside	30	OC-3	Uniform[10 20]	
8-cluster	24	Outside	13	OC-12	Uniform[20 40]	1.541
		Inside	24	OC-3	Uniform[10 20]	

Table 4.1: Metrics for Multiple Cluster Topologies

3-Cluster Network



8-Cluster Network



Performance Evaluation

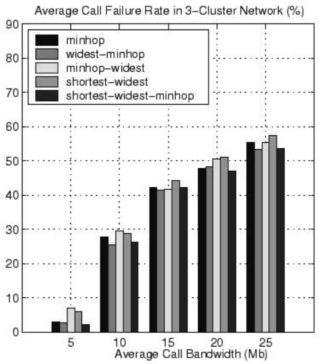
Performance Evaluation

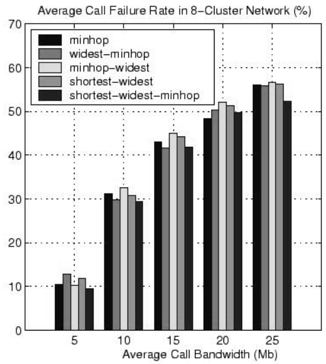
- Maximum bandwidth routing tests
- Minimum delay routing tests
- Link utilization tests
- Alternate routing tests
- Network density tests

Maximum Bandwidth Routing Tests

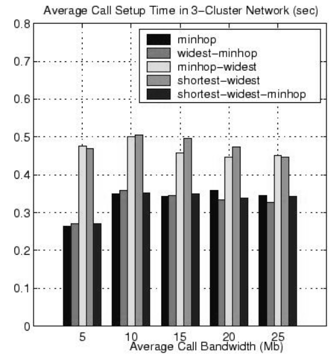
- Calls are CBR-typed with different bandwidth requests
- Call Arrival: 5 seconds between calls with Poisson distribution.
- Call duration: 60 seconds with Poisson distribution.
- Destination Hosts: uniformly selected from all other nodes.
- Total Calls: 2400 calls

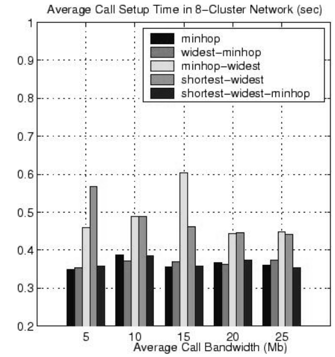
Average Call Blocking Rate and Call Bandwidth



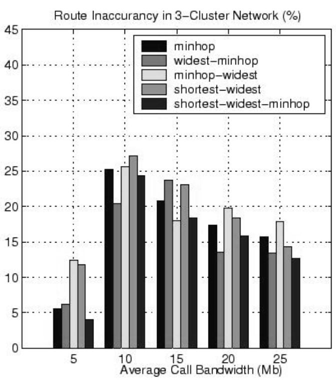


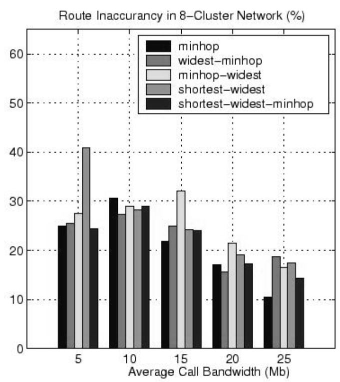
Average Call Setup Time and Call Bandwidth





Routing Inaccuracy and Call Bandwidth



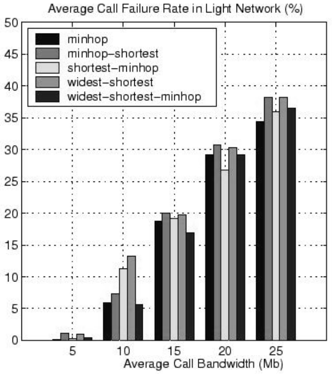


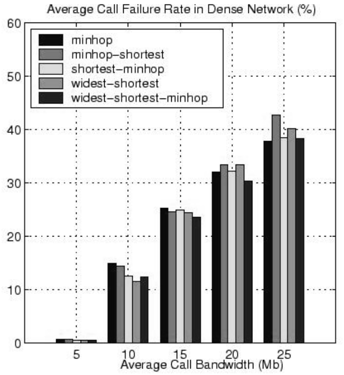
Minimum Delay Routing Tests

Minimum Delay Routing Tests

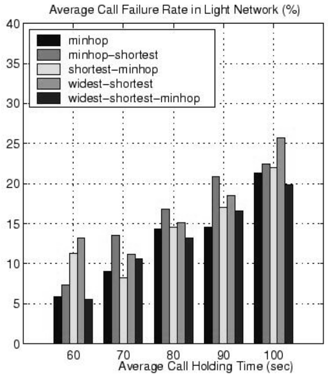
- Calls are CBR-typed with different bandwidth requests
- Call Arrival: 5 seconds between calls with Poisson distribution.
- Call duration: varied with Poisson distribution.
- Destination Hosts: uniformly selected from all other nodes.
- Total Calls: 2400 calls

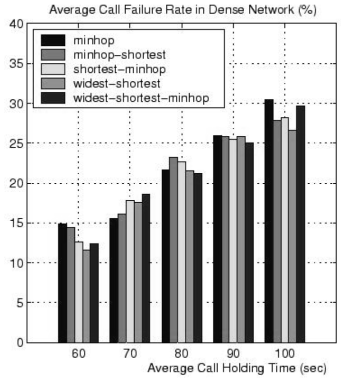
Average Call Blocking Rate and Call Bandwidth





Average Call Blocking Rate and Call Holding Time





Link Utilization Tests

Link Utilization Tests

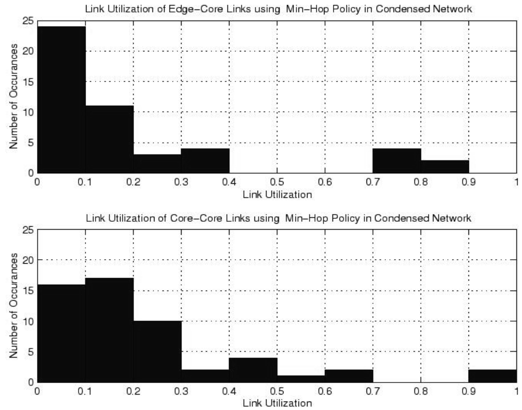
- What is Link Utilization?

$$= \frac{\text{Total BW used of the link}}{\text{Link Capacity}}$$

- Calls are CBR-typed: an average of *uniformly* distributed call bandwidth:10 Mbps.
- Call arrival: 5 seconds between calls with *Poisson* Distribution
- Call duration: 60 seconds with *Poisson* distribution
- One host makes 1000 calls
- Total 24,000 calls in the network.

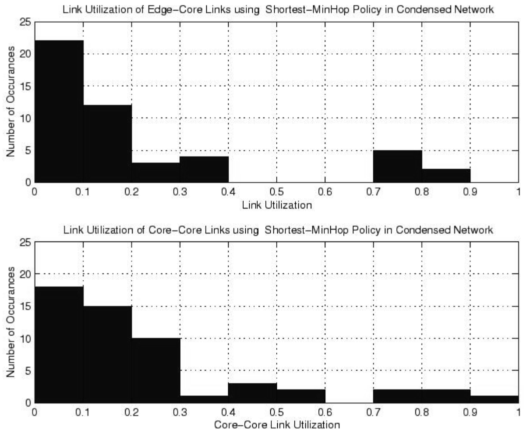
Link Utilization in Edge-Core Network

Minhop



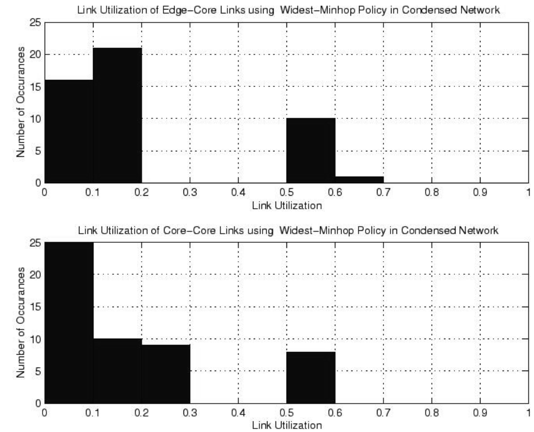
Link Utilization in Edge-Core Network

Shortest-minhop



Link Utilization in Edge-Core Network

Widest-minhop



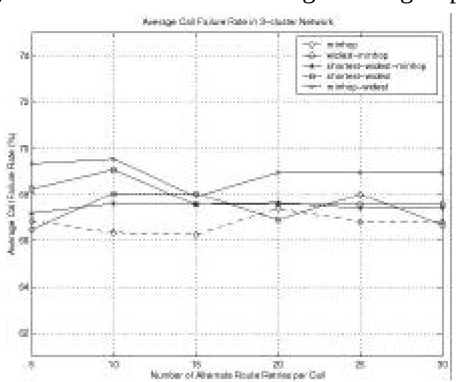
Alternate Routing Tests

Alternate Routing Tests

- Calls are CBR-typed: an average of *uniformly* distributed call bandwidth: 30 Mbps.
- Call arrival: 5 seconds between calls with Poisson distribution.
- Call duration: 60 seconds with Poisson distribution
- Total number of calls: 2400 calls
- We increase the number of alternate routing retries.

Alternate Routing in Cluster Network

- Average call failure rate when using widest group algorithm



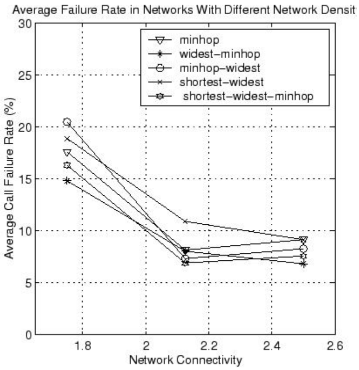
Network Core Density Tests

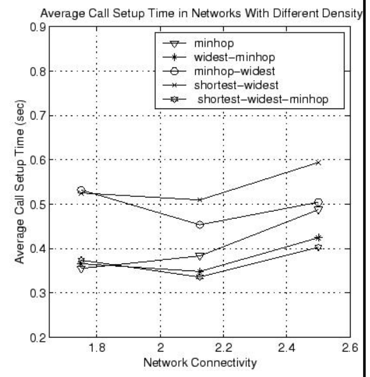
Network Core Density Tests

- Calls are CBR-typed: an average of uniformly distributed call bandwidth: 15 Mbps.
- Call arrivals: 5 seconds between calls with Poisson distribution.
- Call duration: 60 seconds with Poisson distribution.
- Total calls: 2400 calls
- Network density (or connectivity)

Links	Low-dense	Medium-dense	High-dense
Core Links	18	27	36
Edge Links	24	24	24
Nodes	24	24	24
Connectivity	1.75	2.125	2.5

Routing with Different Network Core Density





Conclusions

Conclusions

- In maximum bandwidth routing, the widest-minhop and the shortest-widest-minhop routing algorithms tend to perform better than others in the widest algorithm group.
- However, the minhop-widest and the shortest-widest routing algorithms tend to perform worse than others in the widest algorithm group.
- In minimum delay routing, those algorithms does not perform well because they do not consider the dynamic change of the network.

Conclusions (continued)

- Widest-minhop routing algorithm can improve the link utilization of the network.
- Increasing the number of alternate routing retries slightly improves the call success rate.
- Increasing the number of the core links in the edge-core network improves the call success rate, BUT not always.
- At a certain point, increasing the network density does not reduce the call failure rate. Instead, it increases the call setup time.
- The large amount of resource information can deteriorate the network performance.