CS 218- QoS Routing + CAC

Fall 2003

- M. Gerla et al: “Resource Allocation and Admission Control Styles in QoS DiffServ Networks,” QoS-IP 2001, Rome, Italy.
- A. Dubrovsky, M. Gerla, S. S. Lee, and D. Cavendish,Internet QoS Routing with IP Telephony and TCP Traffic, In Proceedings of ICC 2000.
- L. Breslau et al “Comments on the performance of Measurement Based Admission Control” Infocom 2000

The ingredients of QoS support

- Call Admission Control
- QoS routing
- Policing
- Scheduling

Call Admission Control Styles

Assumptions:

- Intradomain scenario
- Flow Aggregation in Classes (a la DiffServ)
- QoS Routing (Q-OSPF):

(a) traffic and delay measured at routers

(b) link measurements advertised to nodes

(c) sources compute feasible paths

- MPLS used to “pin” the path

1. Resource Allocation CAC

For each call request:

- examine traffic descriptors (rate, loss Prob, Burst Length) and delay Dmax
- compute equiv Bdw and Buffer for each link (Mitra & Elwalid model)
- With Q-OSPF find feasible paths (bdw&buffer)
- using RSVP-like signaling, update the resource allocation along the path

2. Measurement Based CAC

- When a call request comes in, the edge router examines delay and residual bdw measmts advertised for path to destination
- Call admitted/rejected at edge router based on measurements
- No resource allocation/bookkeeping in core routers

3. Hybrid Scheme

So far we have seen:

- **Res All CAC:** enforces determin. bounds, but is too conservative (link utilization); also, bookkeeping required at core routers
- **Measmt CAC:** is more aggressive, no bookkeeping; but, violates QoS constraints

Is there a "middle of the road" approach?

Hybrid CAC (cont)

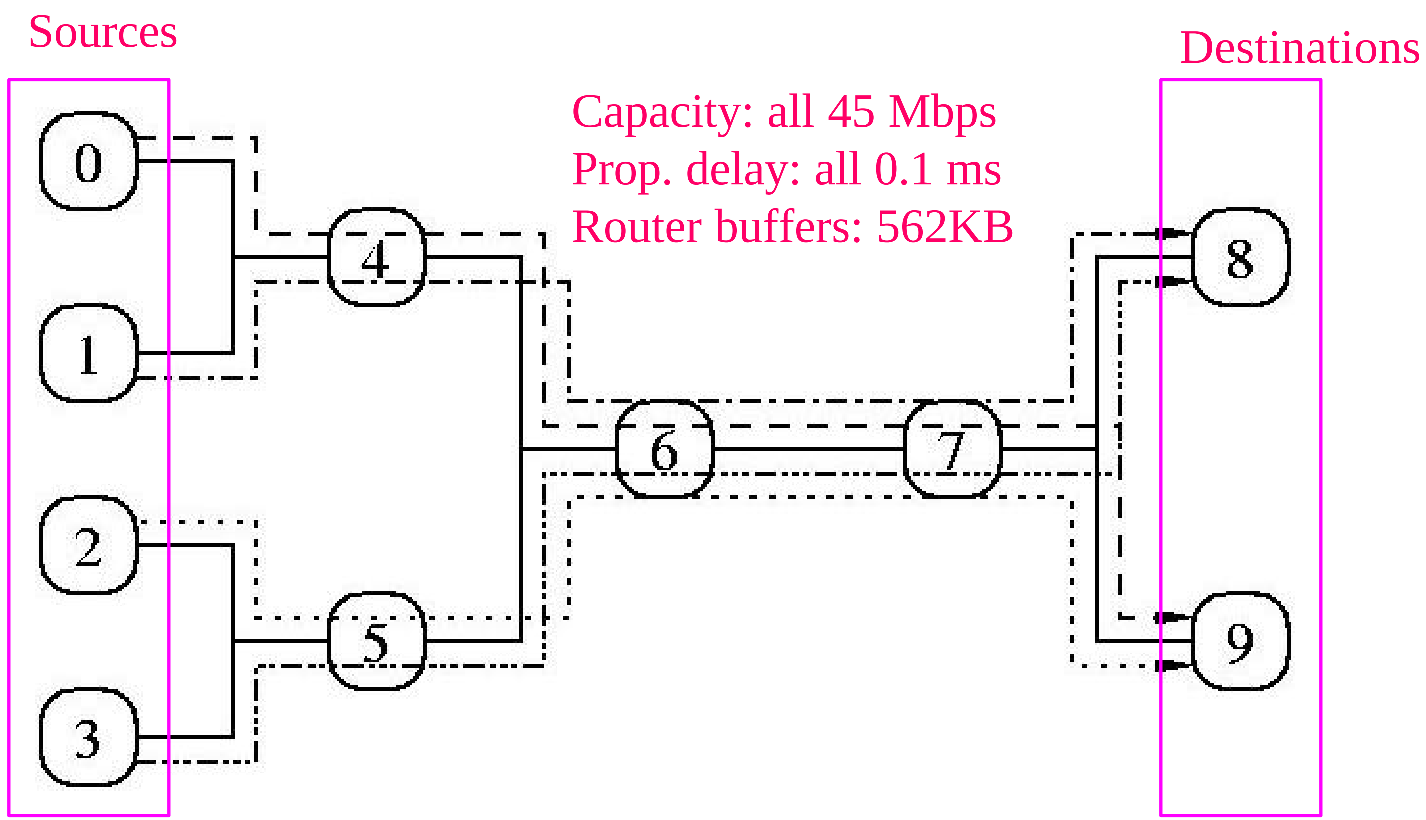
- **Hybrid CAC**:

(a) edge router estimates number of flows (from Q-OSPF trunk traffic measurements)

(b) from number of flows it computes aggregate equiv bdw

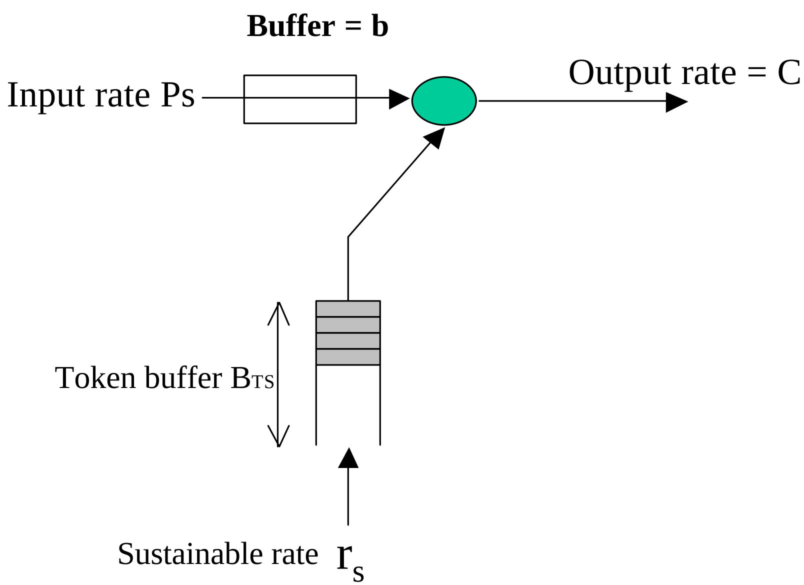
(c) It accepts/rejects call based on Bdw and Buffer availability (no explicit signaling)

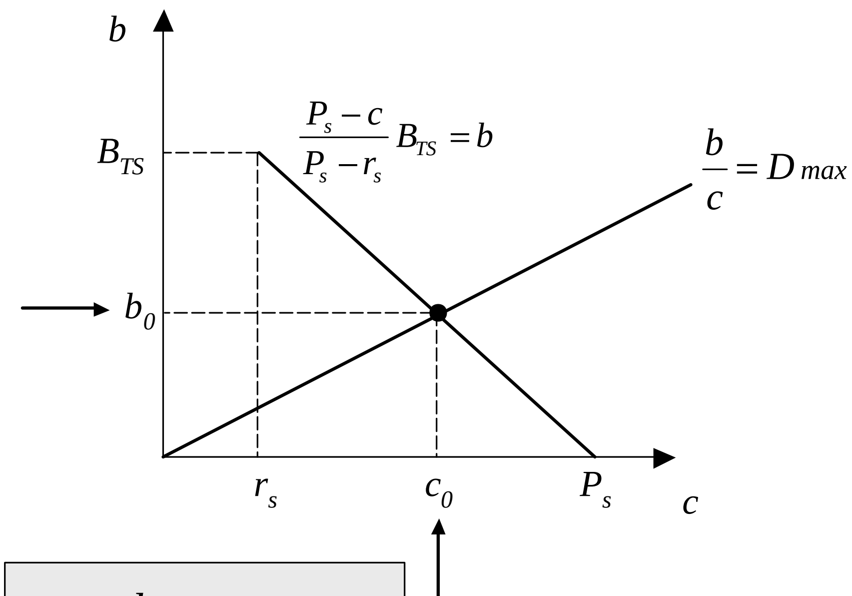
- **Expected result**: performance similar to Res CAC, without core router bookkeeping O/H


Sources
Destinations
Capacity: all 45 Mbps
Prop. delay: all 0.1 ms
Router buffers: 562KB
0
1
2
3
4
5
6
7
8
9

Traffic type	MPEG video trace
Traffic average rate	0.64 Mbps
Traffic peak rate	4.4 Mbps
Connection request arrival	1 per second at each source
Connection duration	60 sec of exponential dist.
Equiv. Bdw allocation	1 Mbps
Equiv. Buffer allocation	12.5 KB

Buffer b and bdw c computation using the LEAKY-BUCKET REGULATOR





b = buffer allocation

$$\frac{P_s - c}{P_s - r_s} B_{TS} = b$$

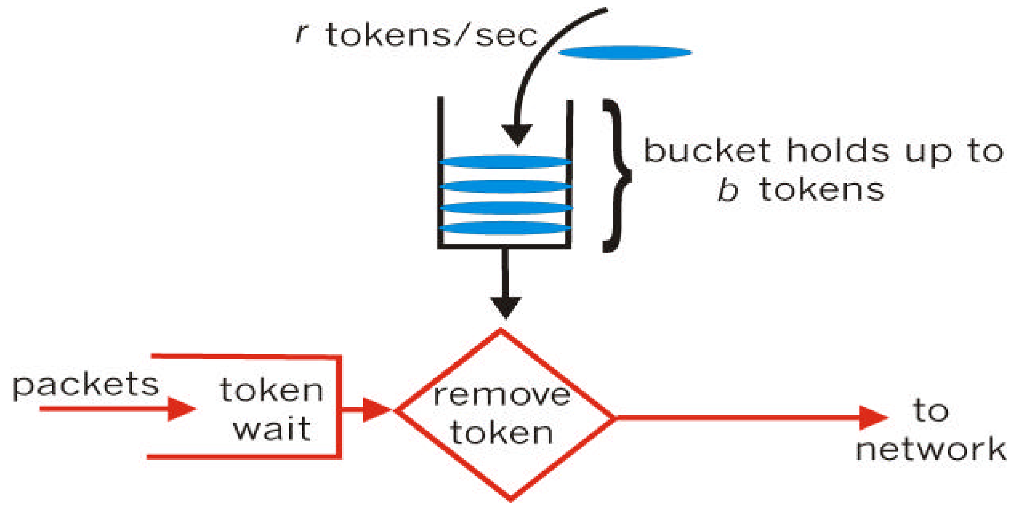
LOSSLESS ALLOCATION

$$\frac{b}{c} = D_{max}$$

SCHEDULING

Policing Mechanism: Token Bucket

Token Bucket: limit input to specified Burst Size and Average Rate.

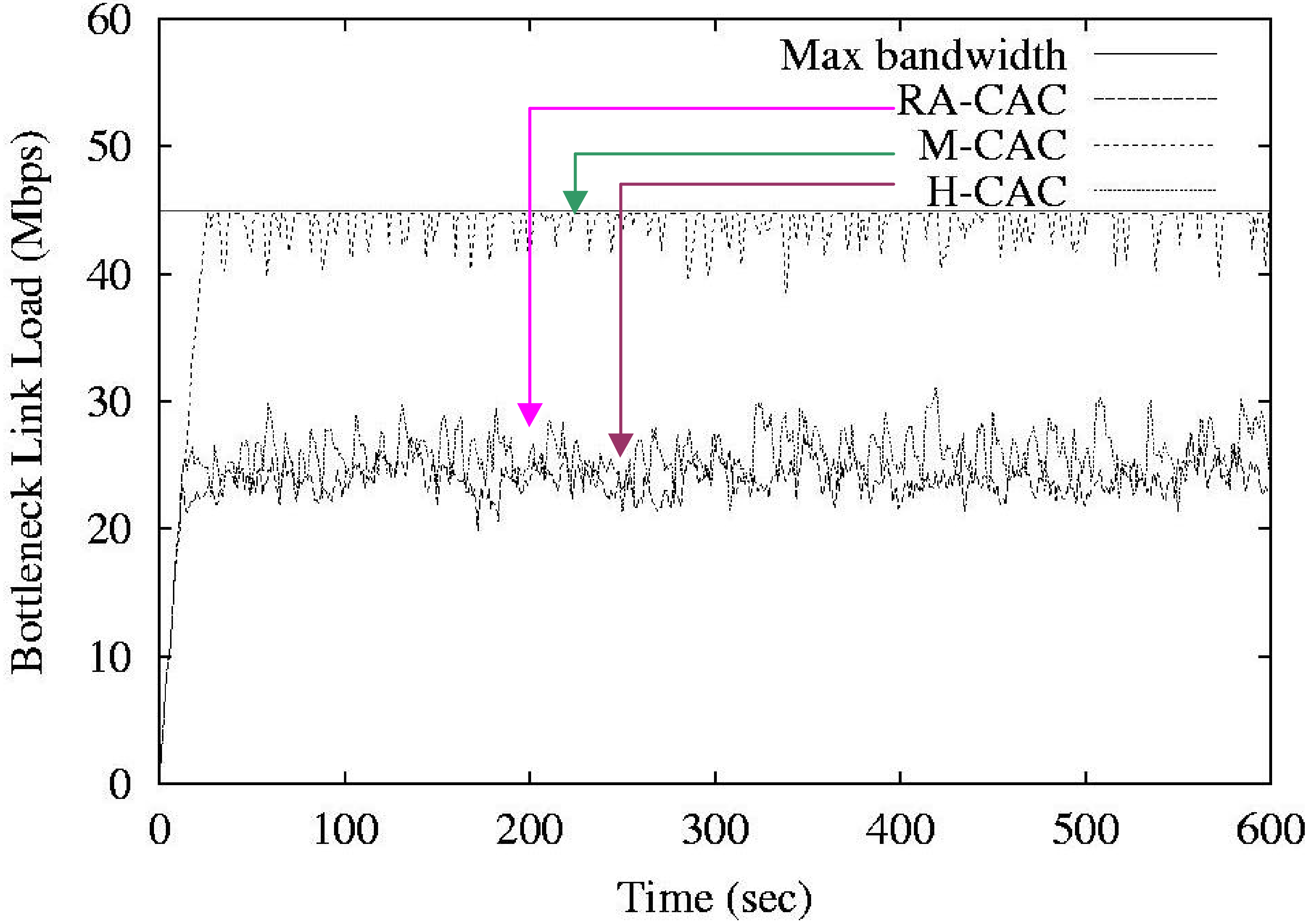


- bucket can hold b tokens
- tokens generated at rate *r token/sec* unless bucket full
- Output removes tokens at channel rate C > r
- Packets arrive at rate < =Ps
- Incoming packet that finds bucket empty is dropped

Sizing equiv buffer *b* and equiv bdw C

- Buffer allocation *b*: must be sufficient to buffer the extra packets during the arrival of the burst B_{ST} (to avoid packet loss)
- Also, delay constraint: $b/C = D_{max}$
- Intersection of the two curves (lossless curve and delay curve) yields optimal $\{\boldsymbol{b_0}, \mathbf{C_0}\}$

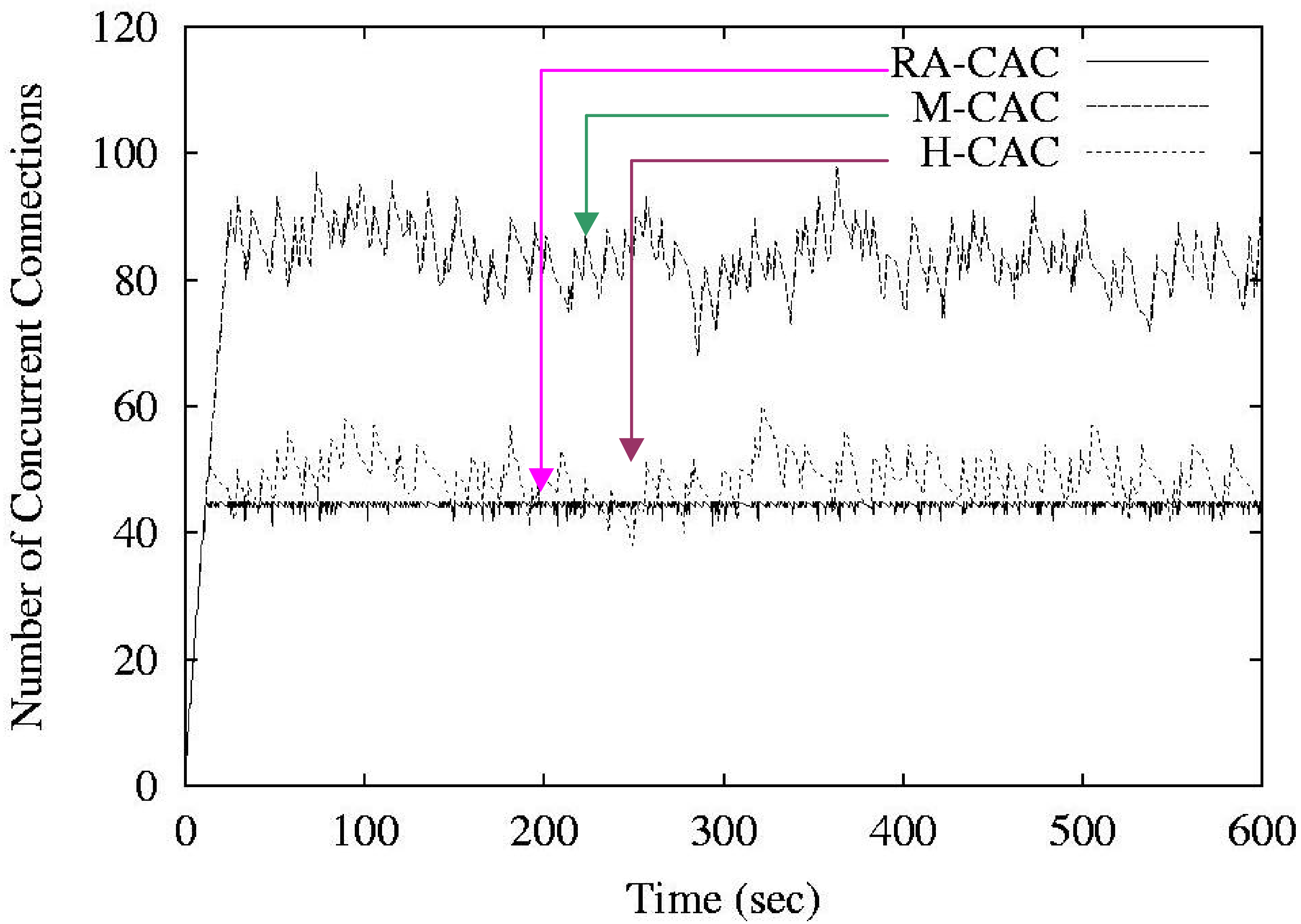
Bottleneck Link Load



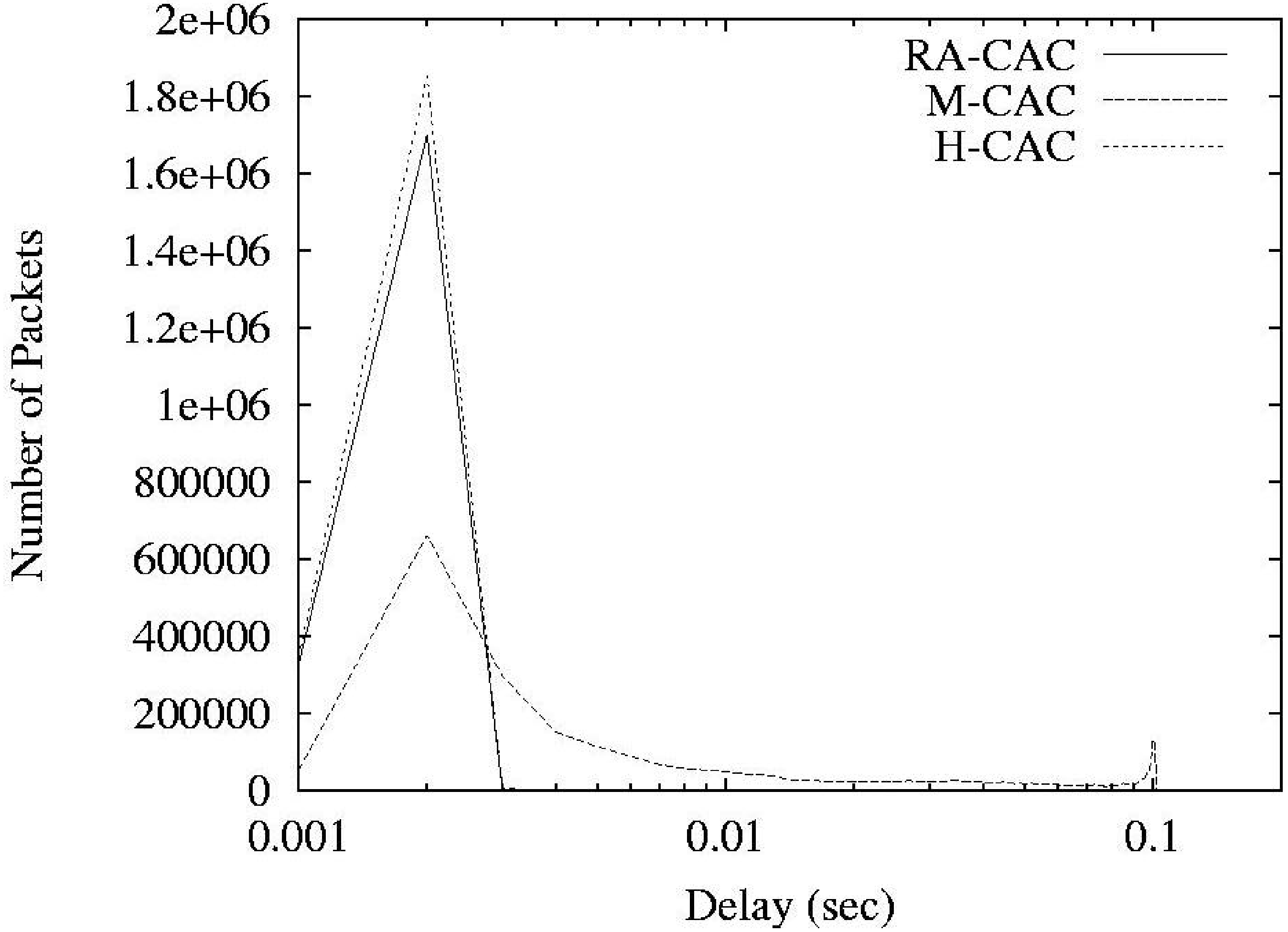
Connections Admitted &Pkt loss

Scheme	# of conn. requests	# of conn. admitted	% of pkt. lost
RA-CAC	2400	465	0 %
M-CAC	2400	864	0.39 %
H-CAC	2400	504	0 %

Connections Admitted



Delay Distribution



CAC Styles: Lessons Learned

- RA-CAC (with determin. bounds) overly conservative (and expensive)
- RA-CAC requires per class “state” at core routers (bdw, buf allocation)
- “state” is drawback in dynamic networks
- “Stateless” options: M-CAC and H-CAC
- Can mix M-CAC and H-CAC (need WFQ)

QoS Routing and Forwarding

Mario Gerla, Gianluca Reali

Scott Lee, Claudio Casetti

Computer Science Department

University of California, Los Angeles (UCLA)

www.cs.ucla.edu/NRL/

Multiple constraints QoS Routing

Given:

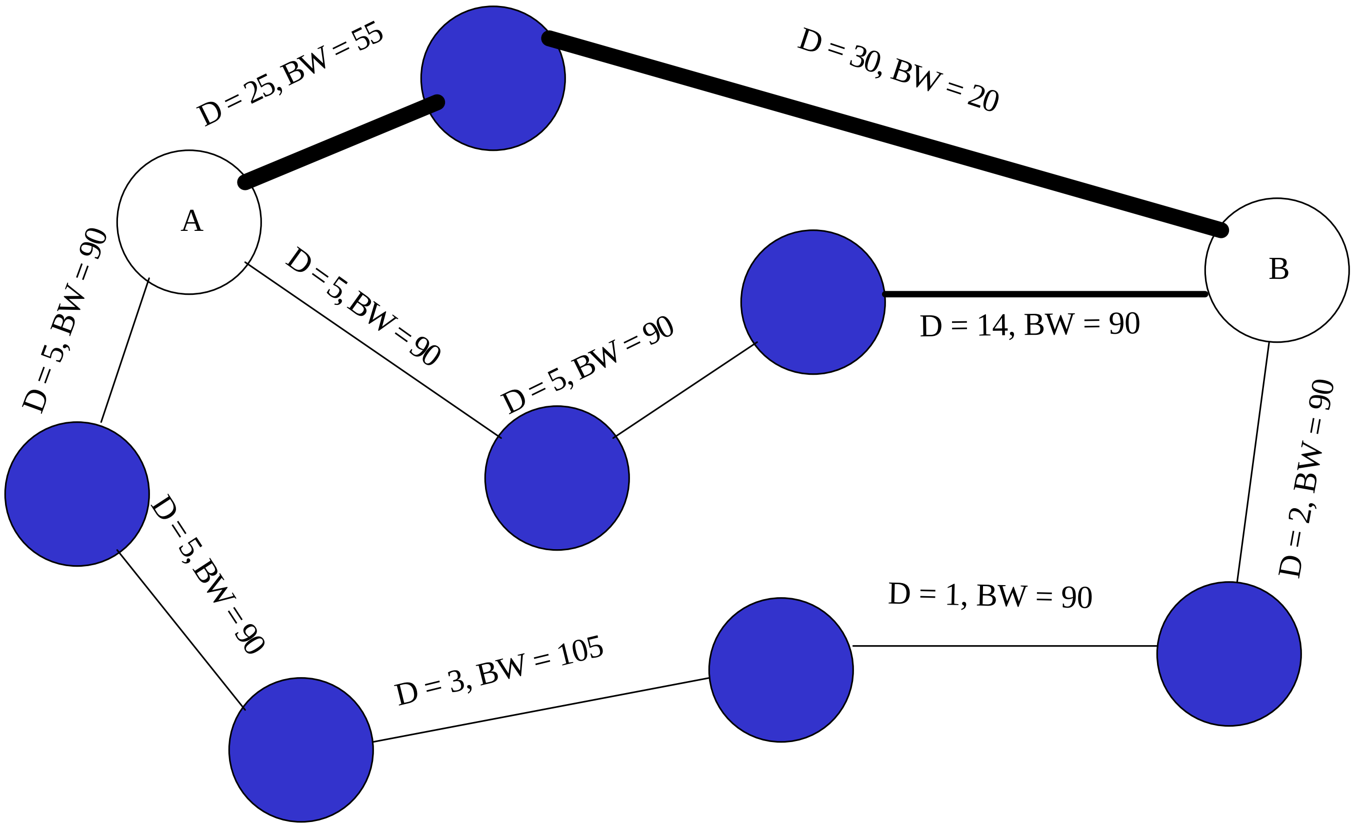
- a (real time) connection request with specified **QoS requirements** (e.g., Bdw, Delay, Jitter, packet loss, path reliability etc)

Find:

- a min cost (typically **min hop**) path which satisfies such constraints

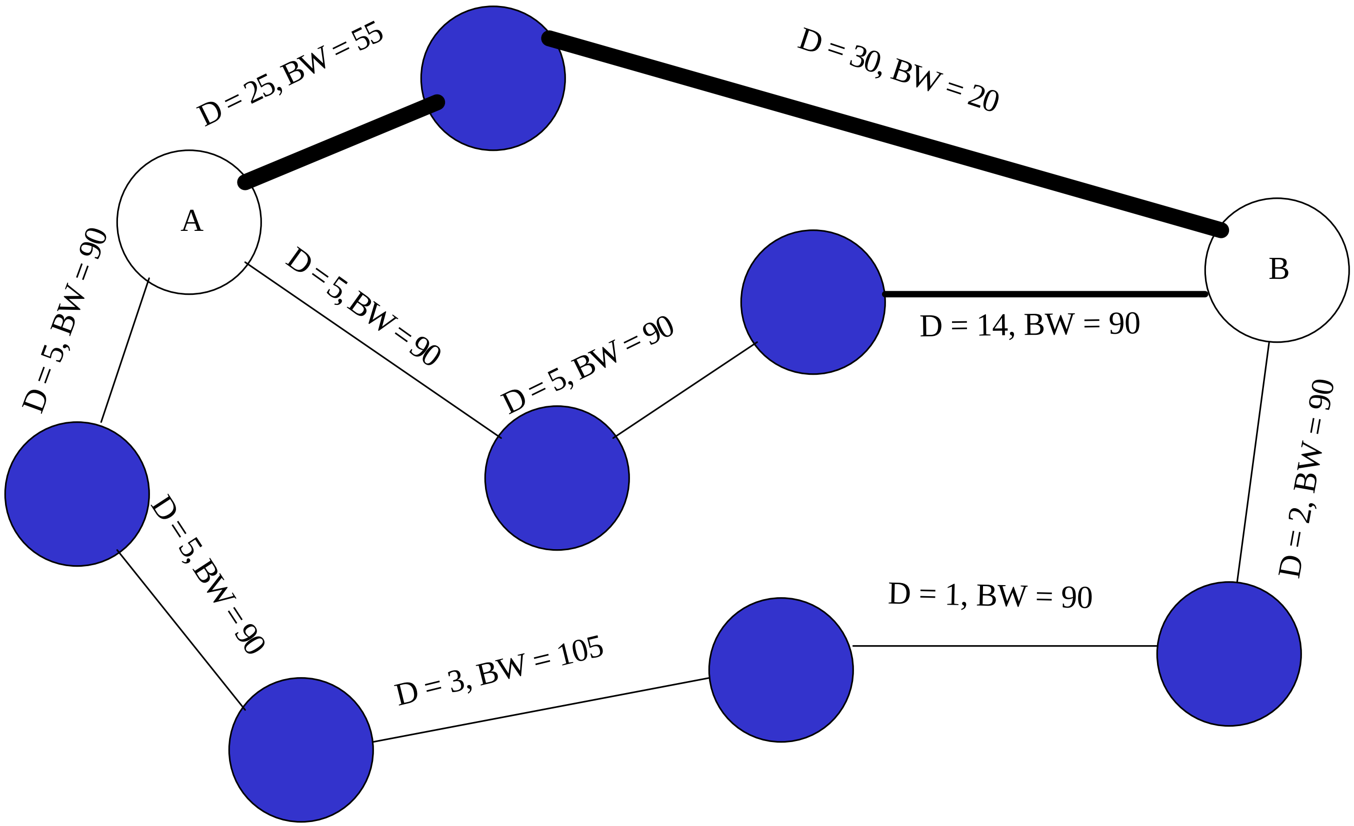
- if no feasible path found, **reject** the connection

Example of QoS Routing



<u>Constraints:</u> Delay (D) <= 25, Available Bandwidth (BW) >= 30

2 Hop Path --------------> Fails (Total delay = 55 > 25 and Min. BW = 20 < 30)
3 Hop Path ----------> Succeeds!! (Total delay = 24 < 25, and Min. BW = 90 > 30)
5 Hop Path ----------> Do not consider, although (Total Delay = 16 < 25, Min. BW = 90 > 30)



<u>Constraints:</u> Delay (D) <= 25, Available Bandwidth (BW) >= 30

We look for <u>feasible</u> path with <u>least number of hops</u>

Benefits of QoS Routing

- **Without QoS routing:**
- must probe path & backtrack; non optimal path, control traffic and processing OH, latency

With QoS routing:

- optimal route; “focused congestion” avoidance
- more efficient Call Admission Control (at the source)
- more efficient bandwidth allocation (per traffic class)
- resource renegotiation easier

The components of QoS Routing

- **Q-OSPF**: link state based protocol; it disseminates link state updates (including QoS parameters) to all nodes; it creates/maintains global topology map at each node
- **Bellman-Ford** constrained path computation algorithm: it computes constrained min hop paths to all destinations at each node based on topology map
- **(Call Acceptance Control**)
- **Packet Forwarding**: source route or MPLS

OSPF Overview

5 Message Types

1) "Hello" - lets a node know who the neighbors are

2) Link State Update - describes sender's cost to its neighbors

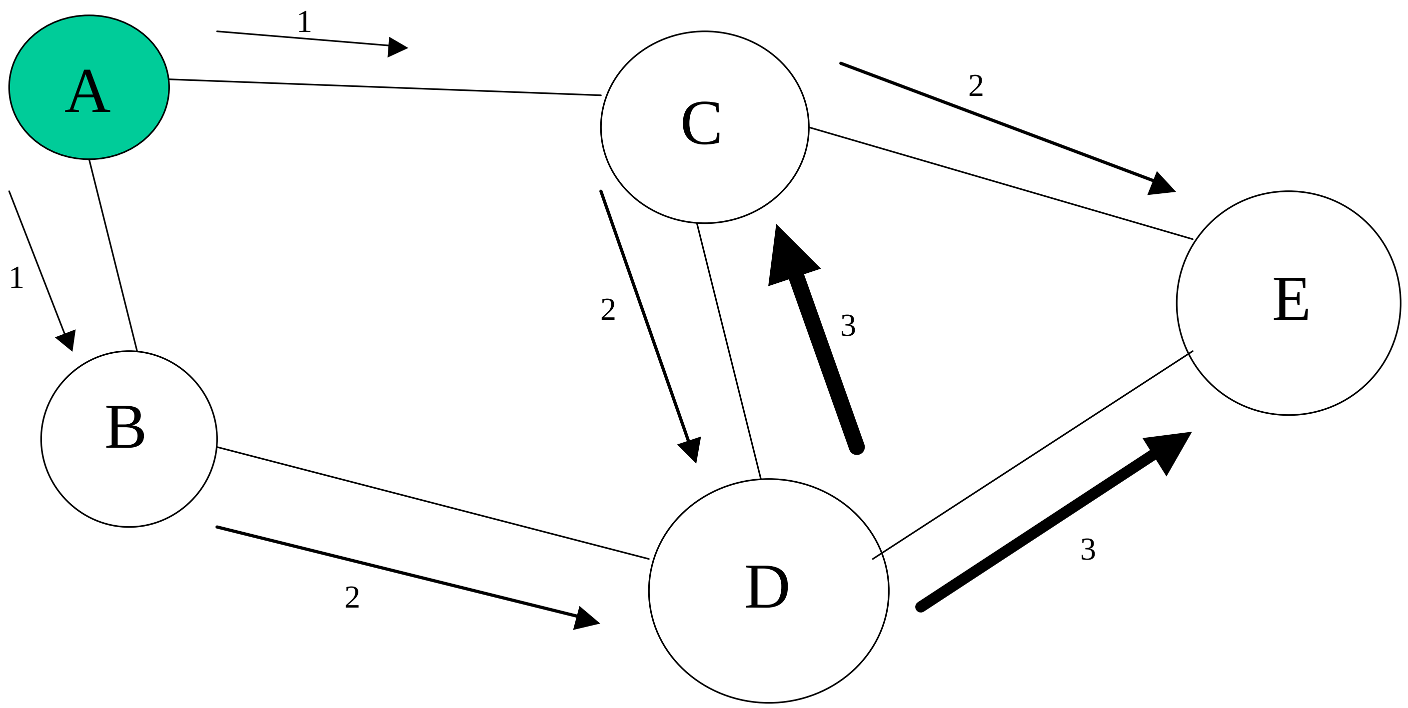
3) Link State Ack. - acknowledges Link State Update

4) Database description - lets nodes determine who has the most recent link state information

5) Link State Request - requests link state information

OSPF Overview(cont)

“Link State Update Flooding”



OSPF Overview (cont)

- “Hello” message is sent every 10 seconds and only between neighboring routers

- Link State Update is sent **every 30 minutes** or upon a change in a cost of a path

- Link State Update is the only OSPF message which is acknowledged

- Routers on the same LAN use “Designated Router” scheme

Implementation of OSPF in the QoS Simulator

- Link State Update is sent **every 2 seconds**
- **No acknowledgement** is generated for Link State Updates
- Link State Update may include (for example):

  - **Queue size** of each outgoing queue (averaged over 10s sliding window)

  - **Avg delay** on each link

  - **Throughput** on each outgoing link (averaged over 10s sliding window)

  - **Total bandwidth** (capacity of the link)

- Source router can use above information to calculate

  - end-to-end delay

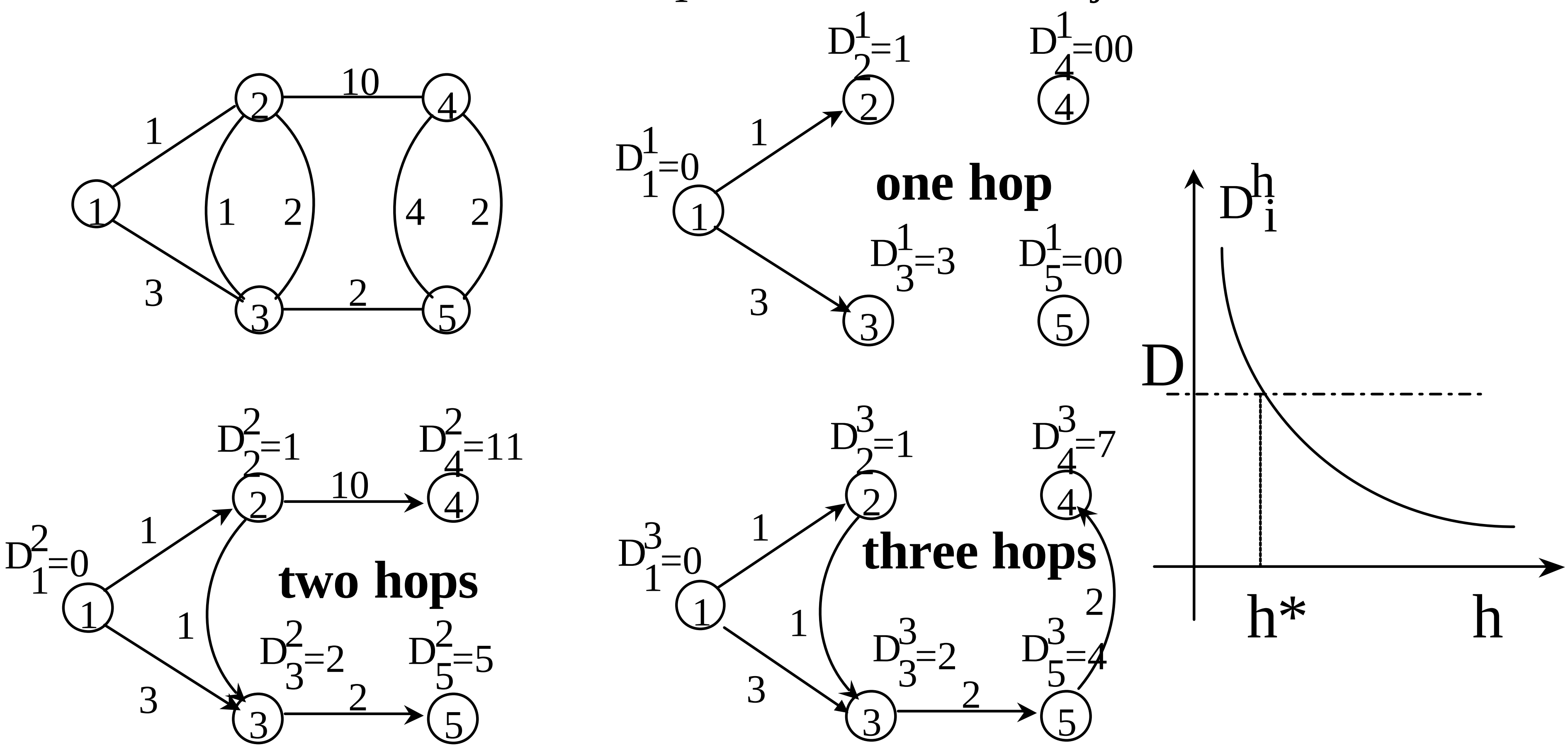
  - available buffer size

  - available bandwidth

Bellman-Ford Algorithm

D_i = delay to node 1 from node i;
d(i,j) = delay of link (i,j); h = iteration number

Bellman Equation : $D_i^{h+1} = \min[\, d(i,j) + D_j^h \,]$



B/F Algorithm properties

- B/F slightly less efficient than Dijkstra (O(NxN) instead of O (NlgN))
- However, B/F generates solutions by increasing hop distance; thus, the first found feasible solution is “hop” optimal (ie, min hop)
- polynomial performance for most common sets of multiple constraints (e.g., bandwidth and delay)

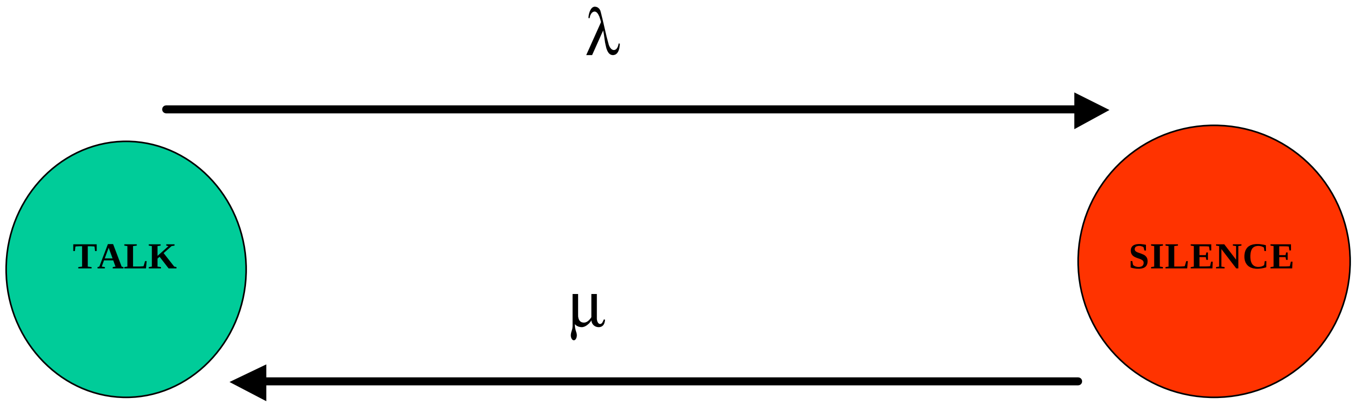
CAC and packet forwarding

- **CAC**: if feasible path not found, call is rejected; alternatively, source is notified of constraint violation, and can resubmit with relaxed constraint (call renegotiation)
- **Packet forwarding**: (a) source routing (per flow), and (b) MPLS (per class)

Application I: IP Telephony

- **M-CAC** at source; **no bandwidth reservation** along path
- 36 node, highly connected network
- Trunk capacity = 15Mbps
- Voice calls generated at fixed intervals
- Non uniform traffic requirement
- Two routing strategies are compared:
  - **Minhop routing (no CAC)**
  - **QoS routing**
- Simulation platform: PARSEC wired network simulation

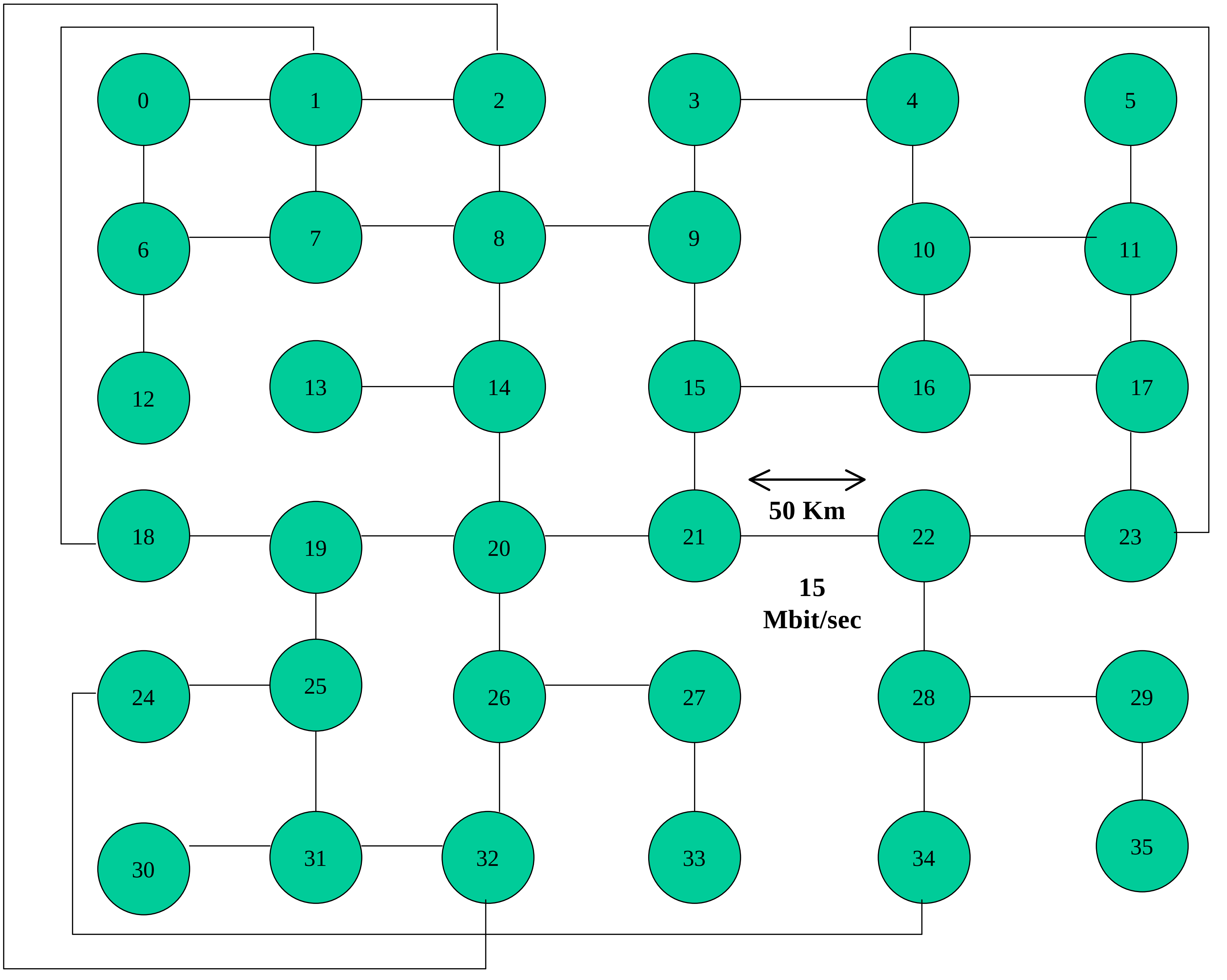
QoS Simulator: Voice Source Modeling



$1/\lambda = 352$ ms

$1/\mu = 650$ ms

1 voice packet every 20ms during talk state


0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
50 Km
18
19
20
21
22
23
15
Mbit/sec
24
25
26
27
28
29
30
31
32
33
34
35

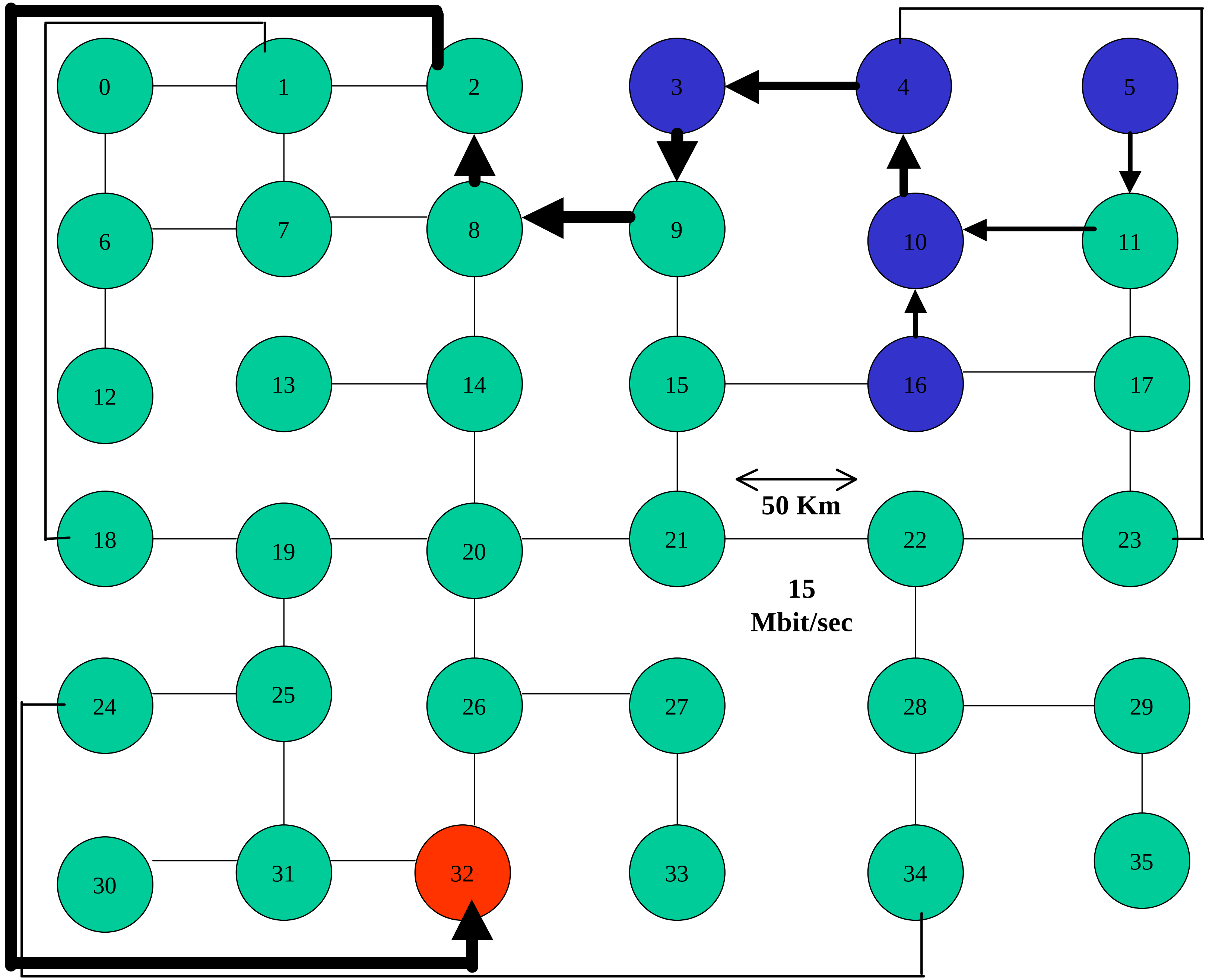
Simulation Parameters

- 10 Minute Simulation Runs
- Each voice connection lasts 3 minutes
- OSPF updates are generated every 2 seconds (30 minute OSPF update interval in Minhop scheme)
- New voice connections generated with fixed interarrival
- Measurements are in STEADY-STATE (after 3 minutes)
- **100 msec delay threshold**
- **3Mbit/sec bandwidth margin on each trunk**
- NON-UNIFORM TRAFFIC GENERATION

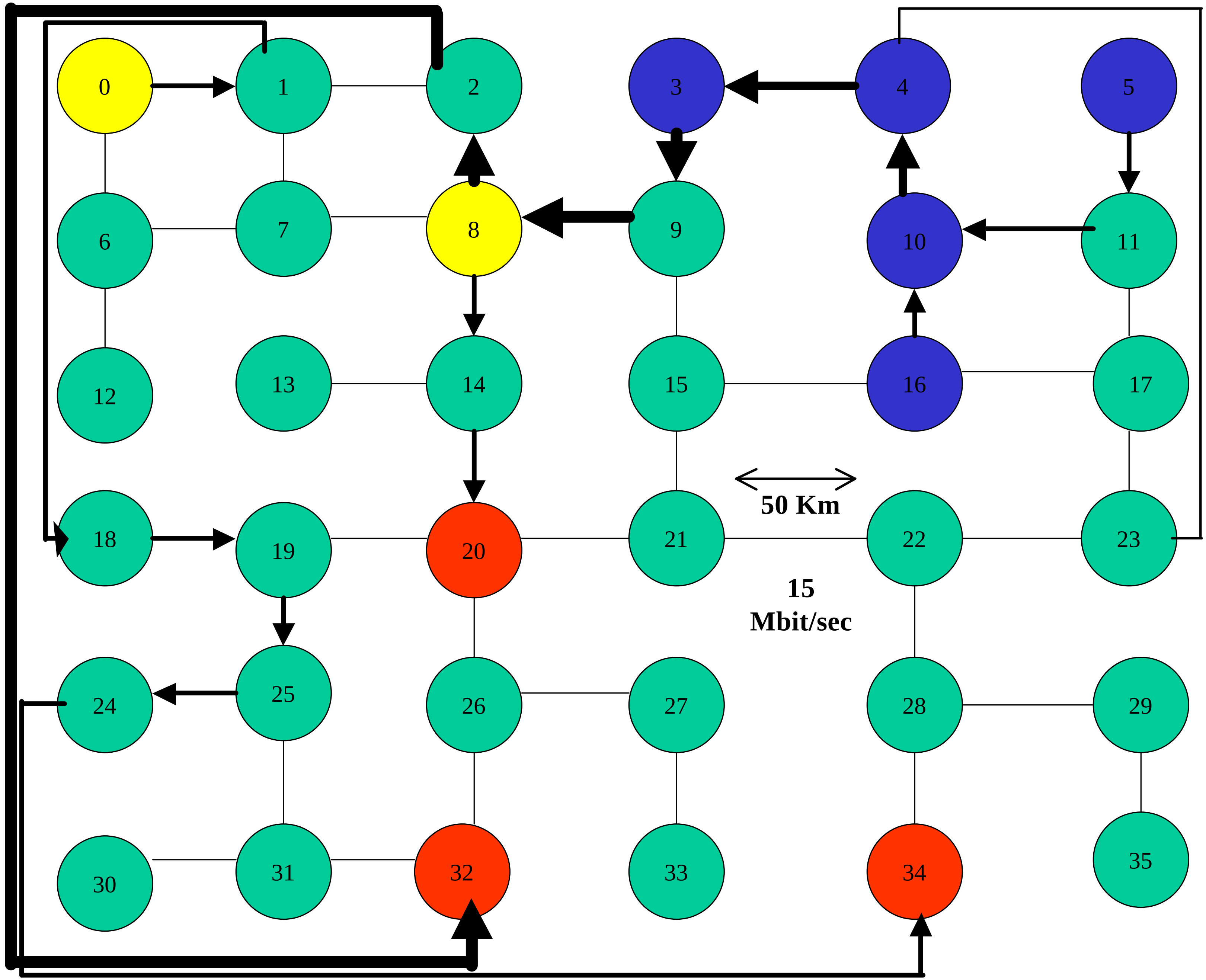
Simulation Results

- The QoS routing accepts all the offered calls by spreading the load on alternate paths

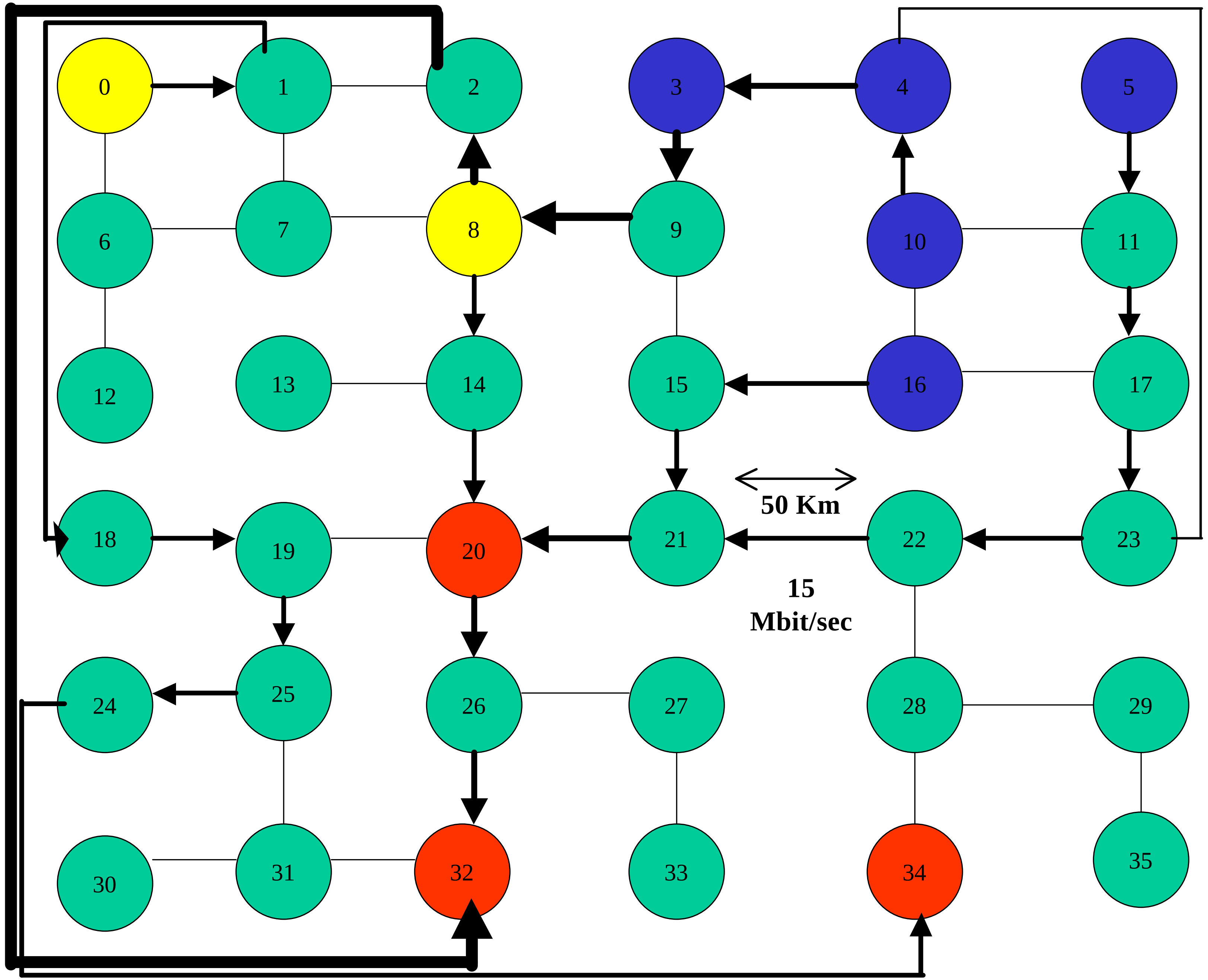
	QoS Routing	Minhop	Minhop w/ CAC
# voice calls attempted in steady state	2762	2762	2790
# voice calls accepted in steady state	2762	2762	1875
% of packets lost	0.0 %	11.78 %	0.0 %
% of packets above 100 ms	0.0 %	51.34 %	0.0 %
% of packets below 100 ms	100.0 %	36.88 %	100.0 %



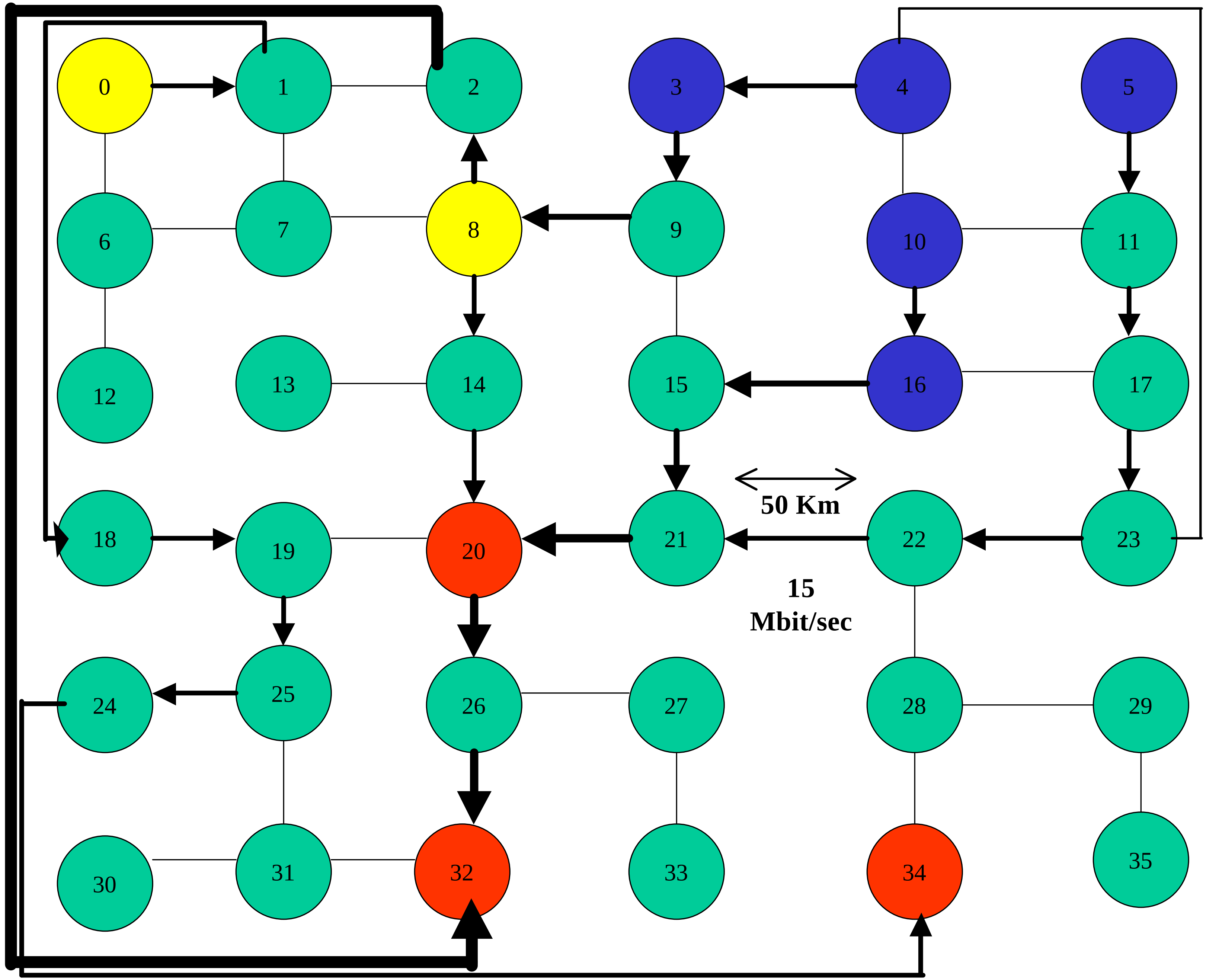
MINHOP ROUTING



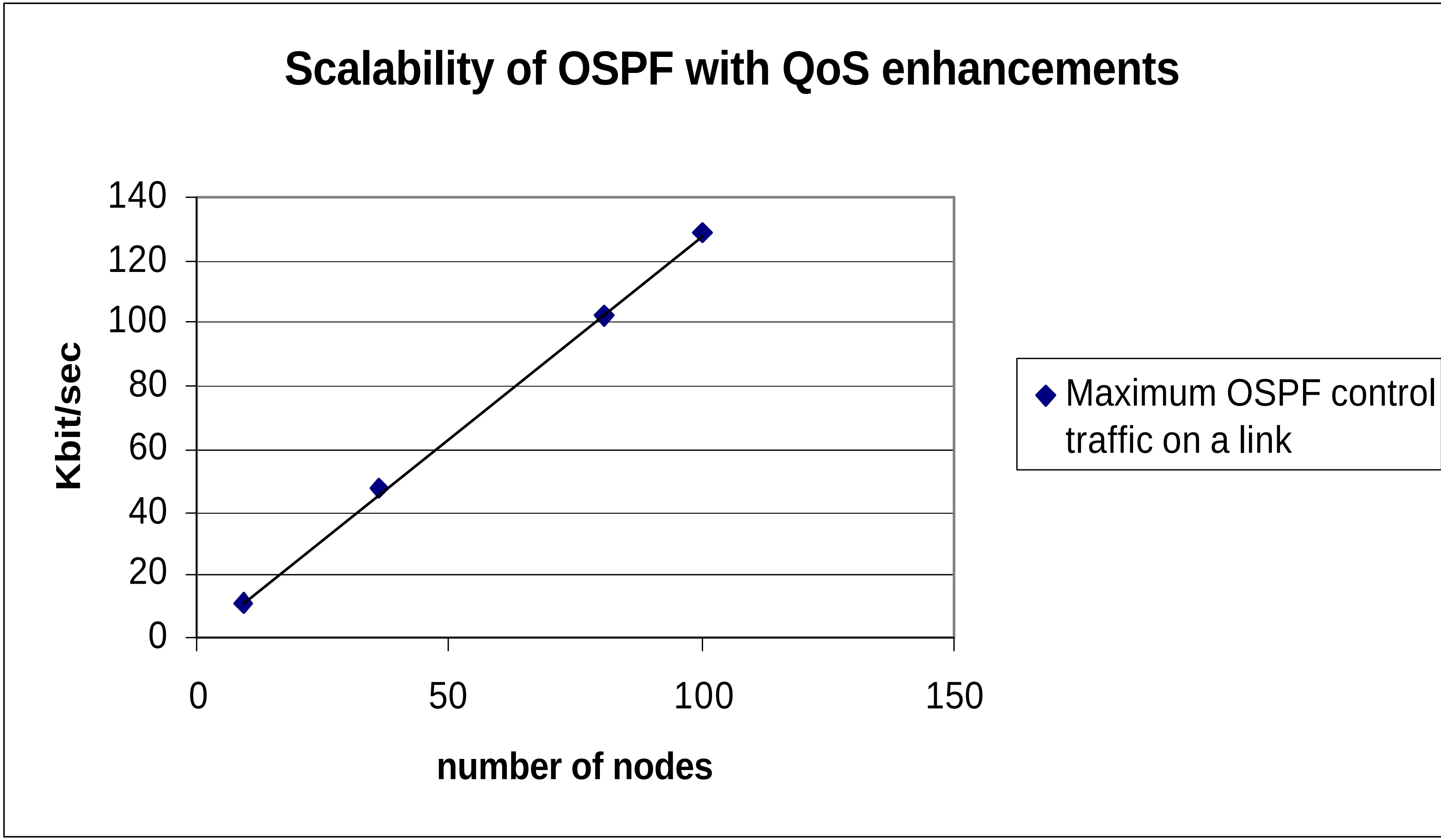
MINHOP ROUTING


0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
50 Km
15 Mbit/sec

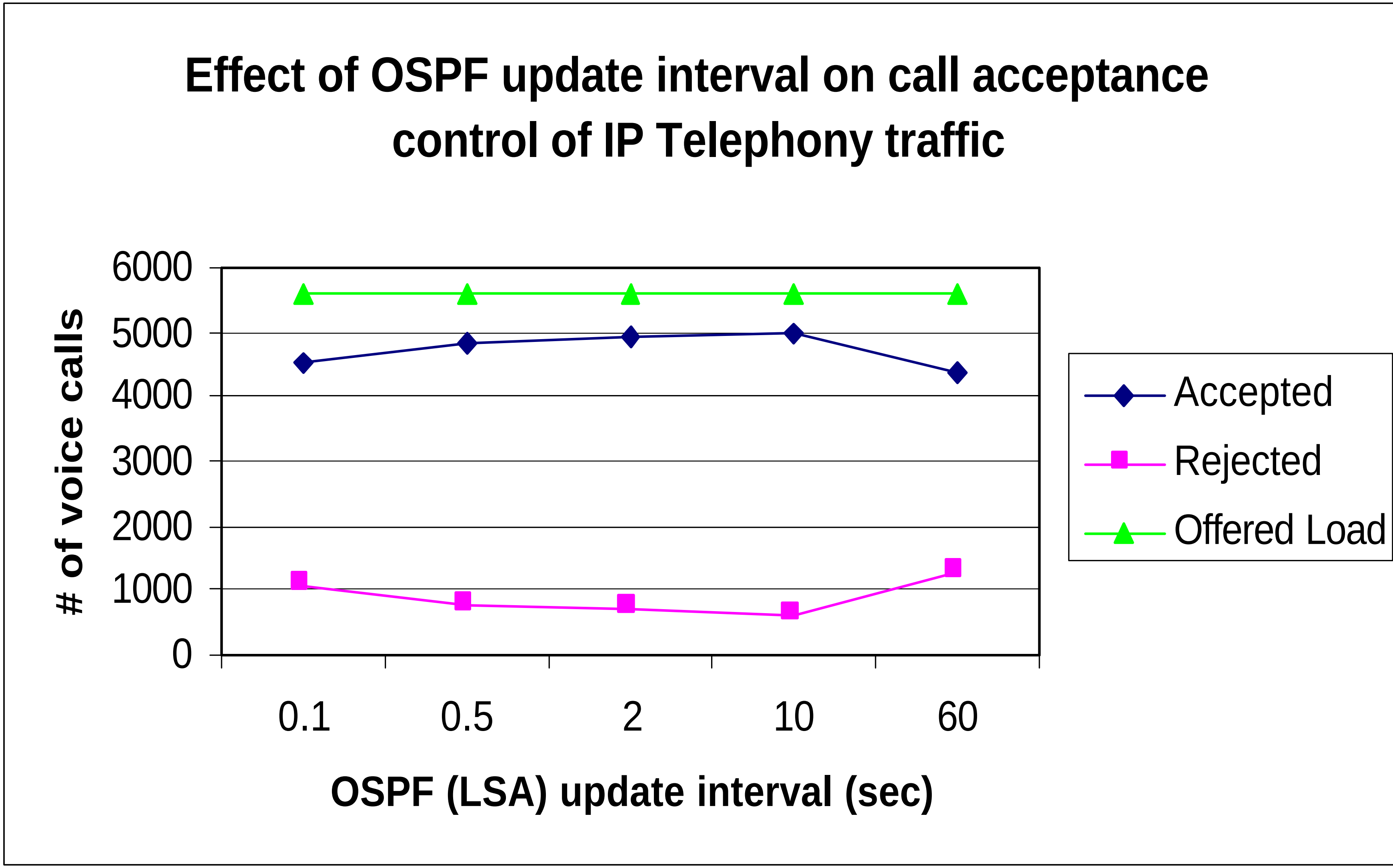
QoS ROUTING


0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
50 Km
15
Mbit/sec

QoS ROUTING



- **OSPF packet size was 350 bytes**
- **OSPF (LSA) updates were generated every 2 seconds**
- **Measurements were performed on a “perfect square grid” topology**



75ms voice call generation rate

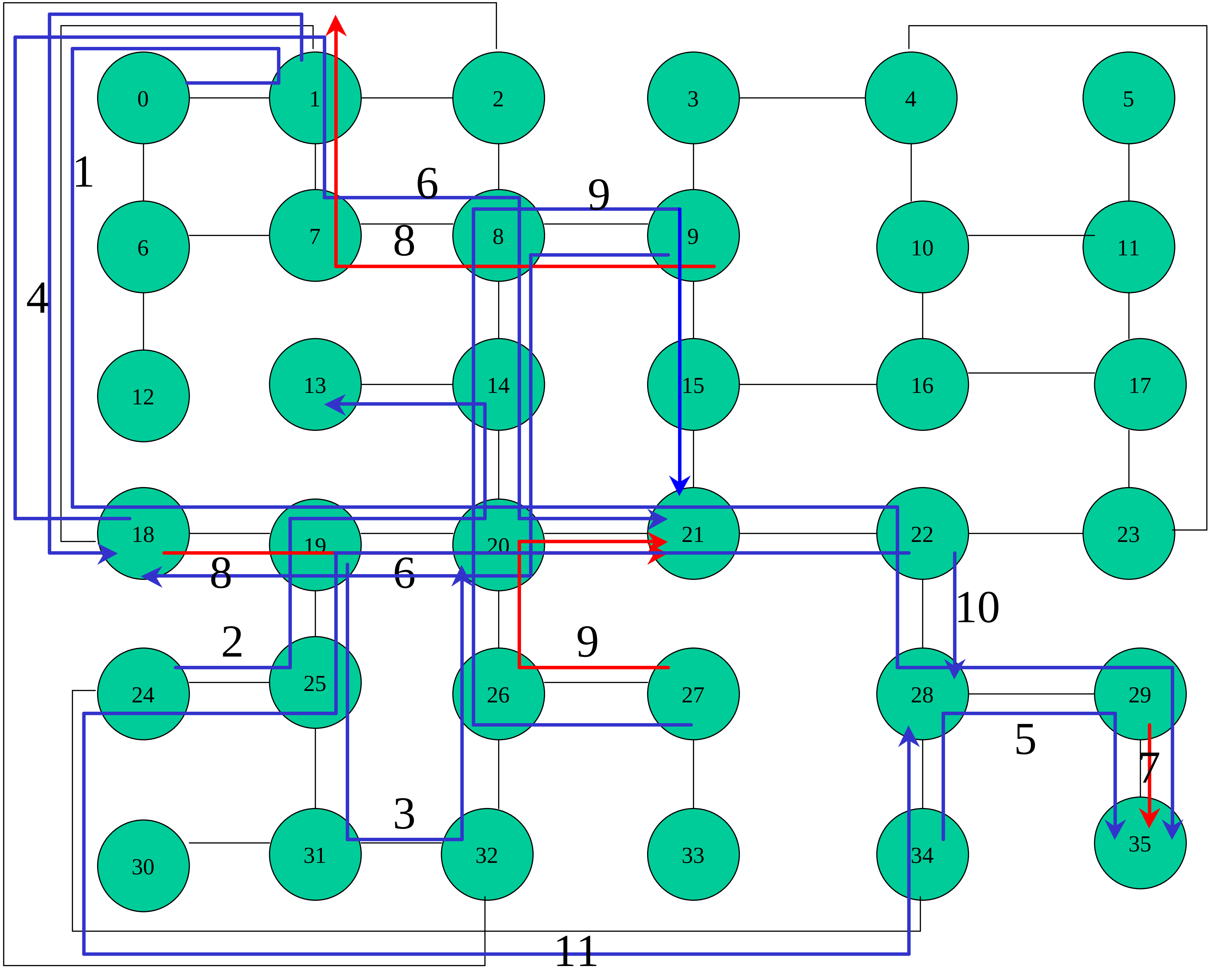
Application II: MPEG video

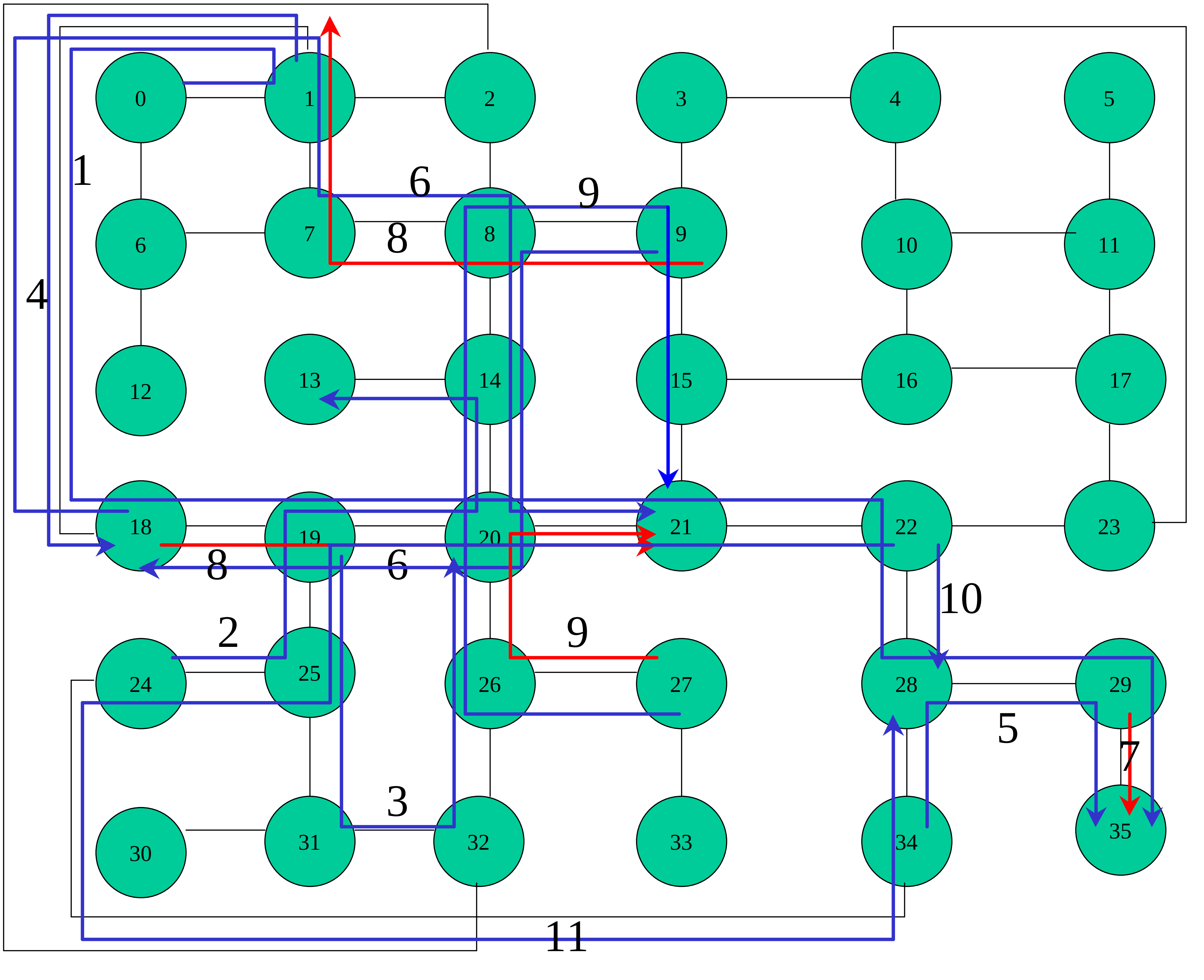
- **Resource Allocation** CAC
- **RSVP type signaling required**
- Effective bandwidth & buffer reservations
- 36 node grid-type topology
- Trunk capacity = 5.5 Mbps
- Inputs = Measured MPEG traces
- QoS guarantees: no-loss; delay < Tmax
- Simulation platform: PARSEC wired network simulation

SIMULATION RESULTS

Bandwidth/link: 5.5 Mbps unidirectional
Tmax: 0.1 s
Effective bandwidth: 2.6 Mbps
Effective buffer: 260 KB (no buffer saturation)

APPLICATION ORDER	SOURCE NODE	DESTINATION NODE	PATH	CAC Y/N	REJECTION NODE
1	0	35	1 18 19 20 21 22 28 29 35	Y	
2	24	13	25 19 20 14 13	Y	
3	19	20	25 31 32 26 20	Y	
4	1	18	18	Y	
5	34	35	28 29 35	Y	
6	18	21	19 20 21	N	19
			1 7 8 14 20 21	Y	
7	29	35	NO PATH FOUND		
8	9	18	8 7 1 18	N	1
			8 14 20 19 18	Y	
9	27	21	26 20 21	N	20
			26 20 14 8 9 15 21	Y	
10	22	28	28	Y	
11	22	28	21 20 19 25 24 34 28	Y	


0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
1
4
6
8
9
8
6
2
9
10
3
5
7
11



0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
1
4
6
8
9
8
6
2
9
10
3
5
7
11

Video Only Result Comparisons

Class based QoS routing with reservation

vs. Measurement based QoS routing without reservation

Bandwidth/link: 5.5 Mbps unidirectional

Tmax: 0.1 s, Duration 10 min

	Class Based QoS routing with reservation	Measurement based QoS routing w/o reservation
Number of packets sent	551820	607002
Percentage of packets lost	0%	0%
Percentage of packets received:	100%	100%
Max delay for video packets:	0.0806 s	0.3725 s
Percentage of Packets exceeding delay threshold	0%	0.8%
Number of connection requests	11	11
Number of rejections	1	0
Number of routing retries	3	N/A

Conclusions

- QoS routing beneficial for CAC, enhanced routing, resource allocation and resource renegotiation
- Can efficiently handle flow aggregation (Diff Serv)
- Q-OSPF traffic overhead manageable up to hundreds of nodes
- Can be scaled to thousands of nodes using hierarchical OSPF
- Major improvements observed in handling of IP telephony and MPEG video
- MPEG video best served via reservations

Future Work

- extension to hierarchical OSPF
- extension to Interdomain Routing
- extension to multiple classes of traffic
- Statistical allocation of MPEG sources