IP DiffServ Architecture for Providing Voice/Video QoS Guarantees: CAC, QoS Routing and Forwarding

Mario Gerla and Gianluca Reali

Computer Science Department

University of California, Los Angeles (UCLA)

gerla@cs.ucla.edu

www.cs.ucla.edu/NRL/

Multiple constraints QoS Routing

Given:

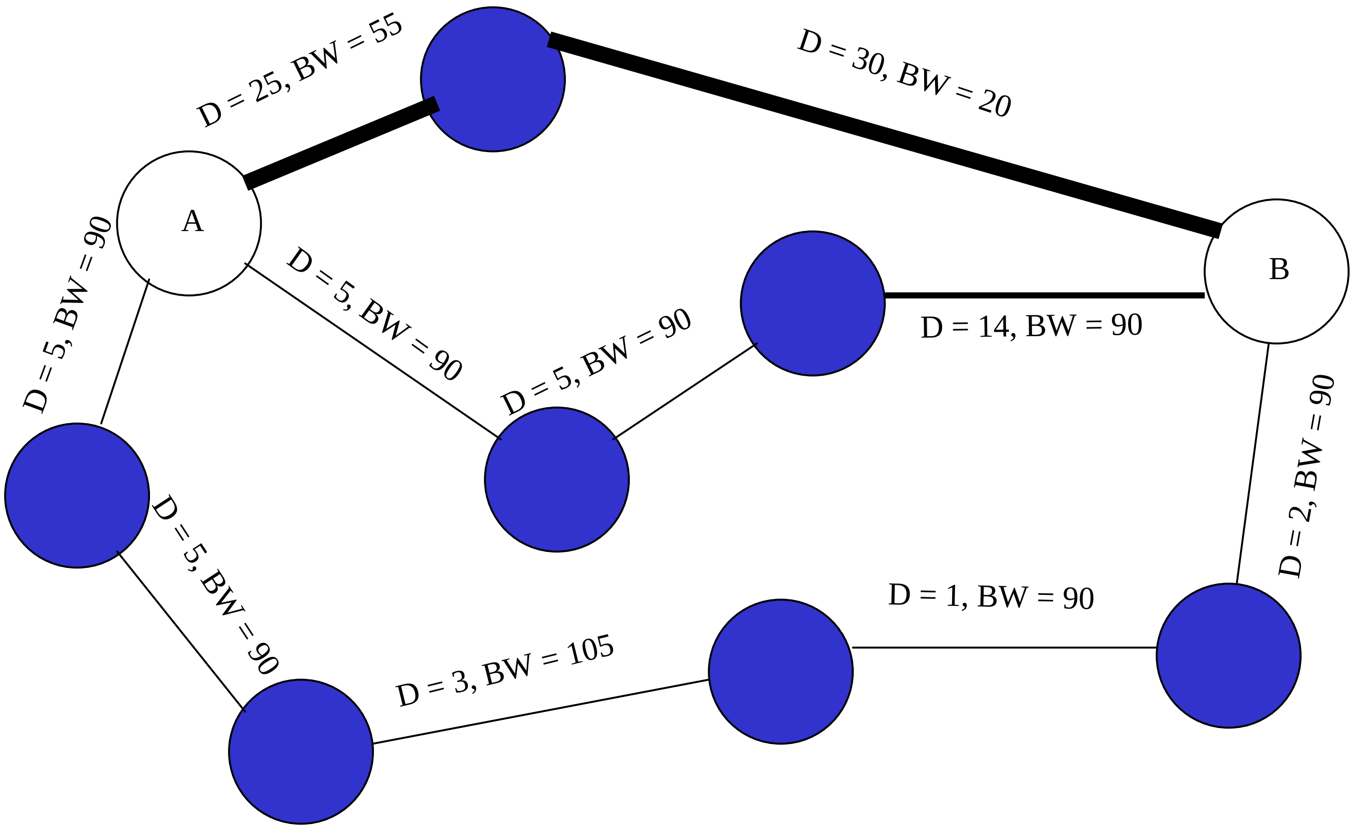
- a (real time) connection request with specified **QoS requirements** (e.g., Bdw, Delay, Jitter, packet loss, path reliability etc)

Find:

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

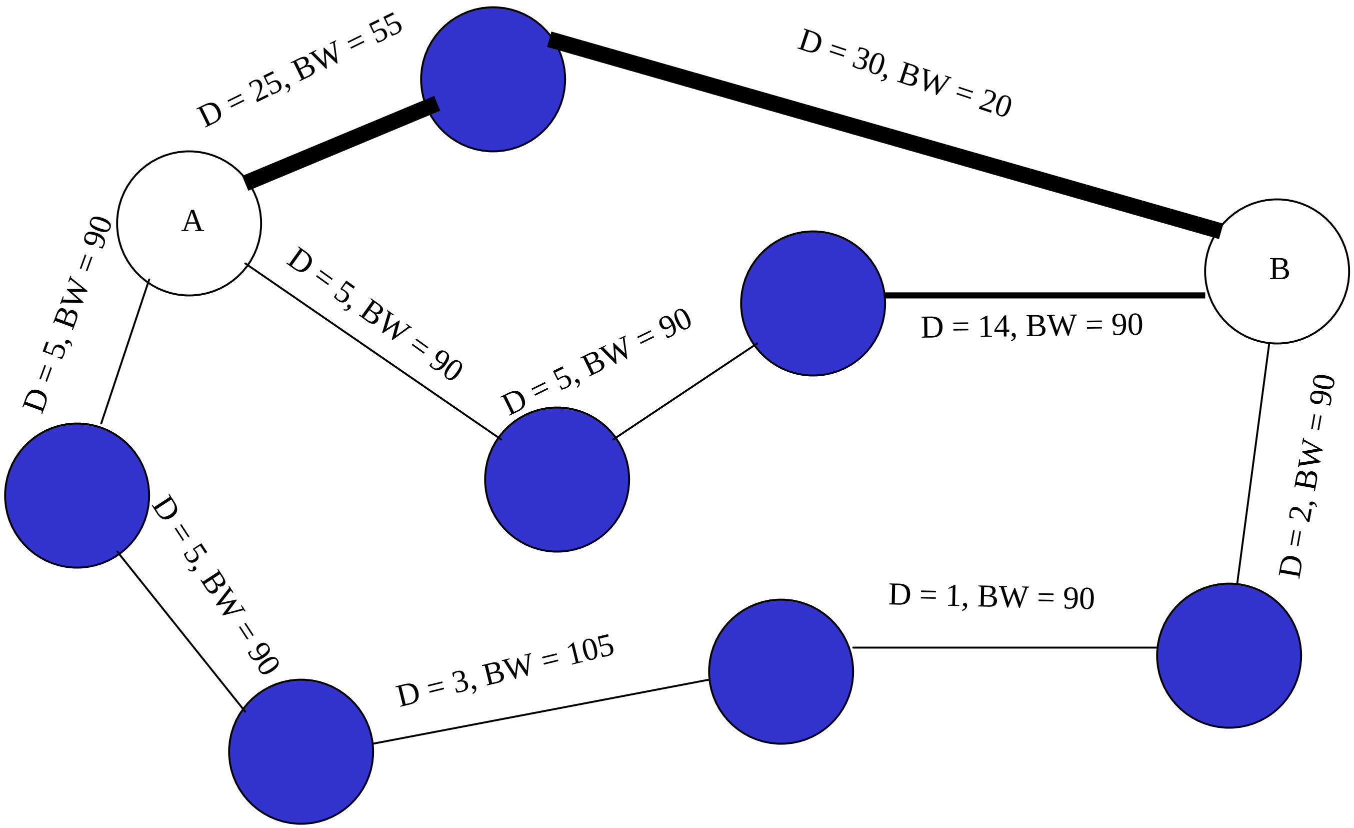
- if no feasible path found, **reject** the connection

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 possible

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 it’s neighbors

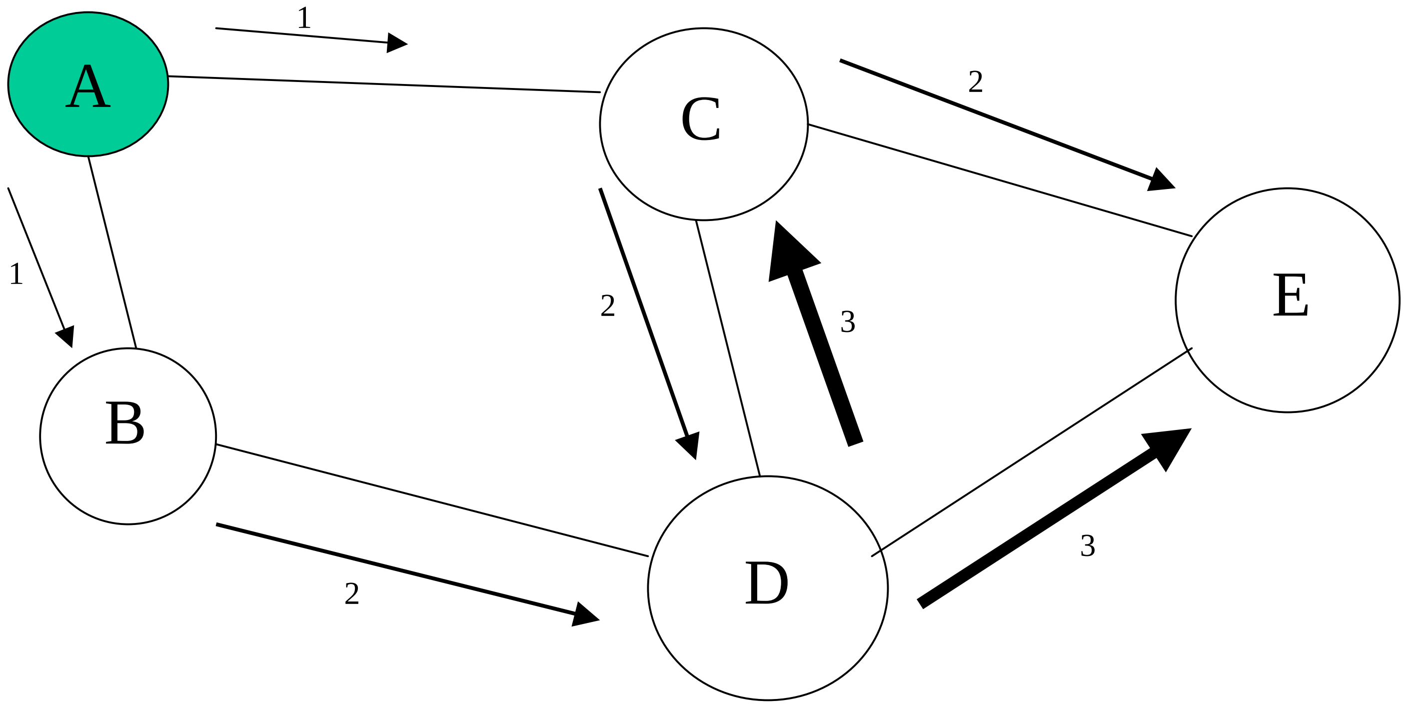
3) Link State Ack. - acknowledges Link State Update

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

5) Link State Request - requests link state information

OSPF Overview

"Link State Update Flooding"



OSPF Overview

- "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
- Information included in each Link State Update:

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

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

  - Total bandwidth (capacity of the link)
- Source router can use above information to calculate

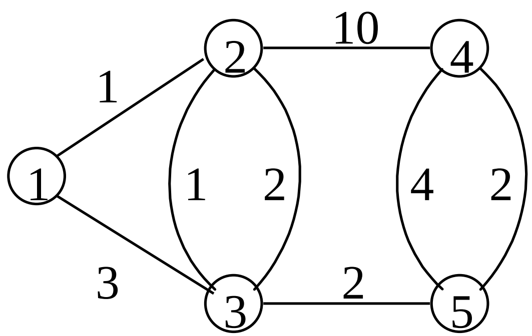
  - end-to-end delay

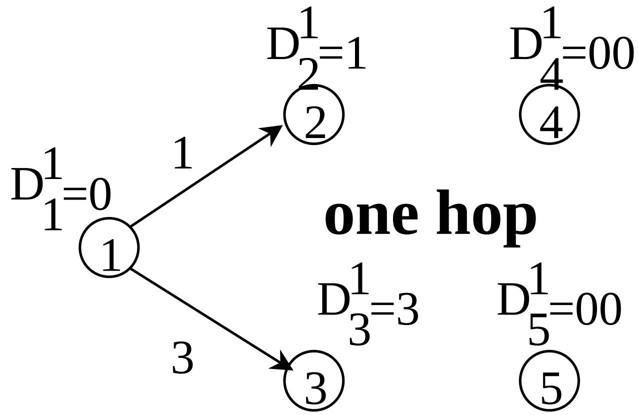
  - available buffer size

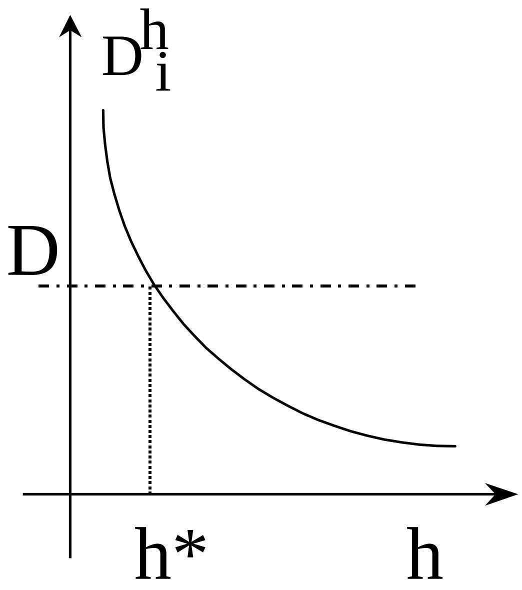
  - available bandwidth

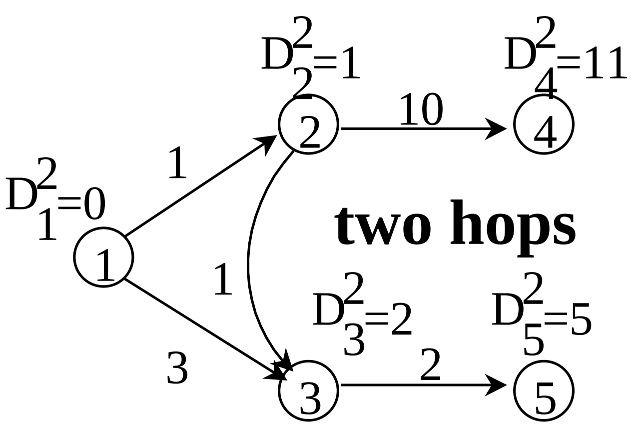
Bellman-Ford Algorithm

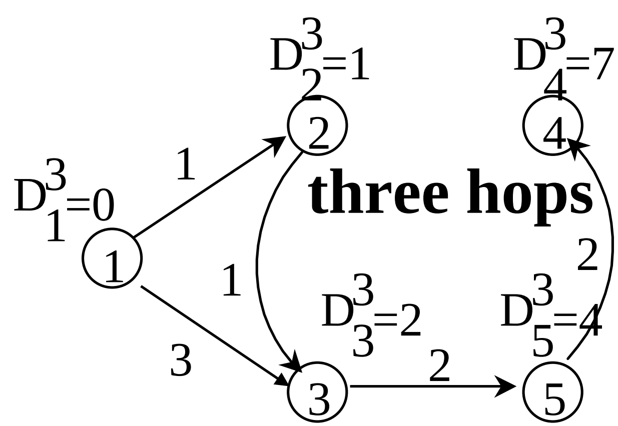
Bellman Equation : $D_i^{h+1} = \min[\, d(i,j) + D_j^h \,]$











B/F Algorithm properties

- B/F slightly less efficient than Dijkstra (O(N2) 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)

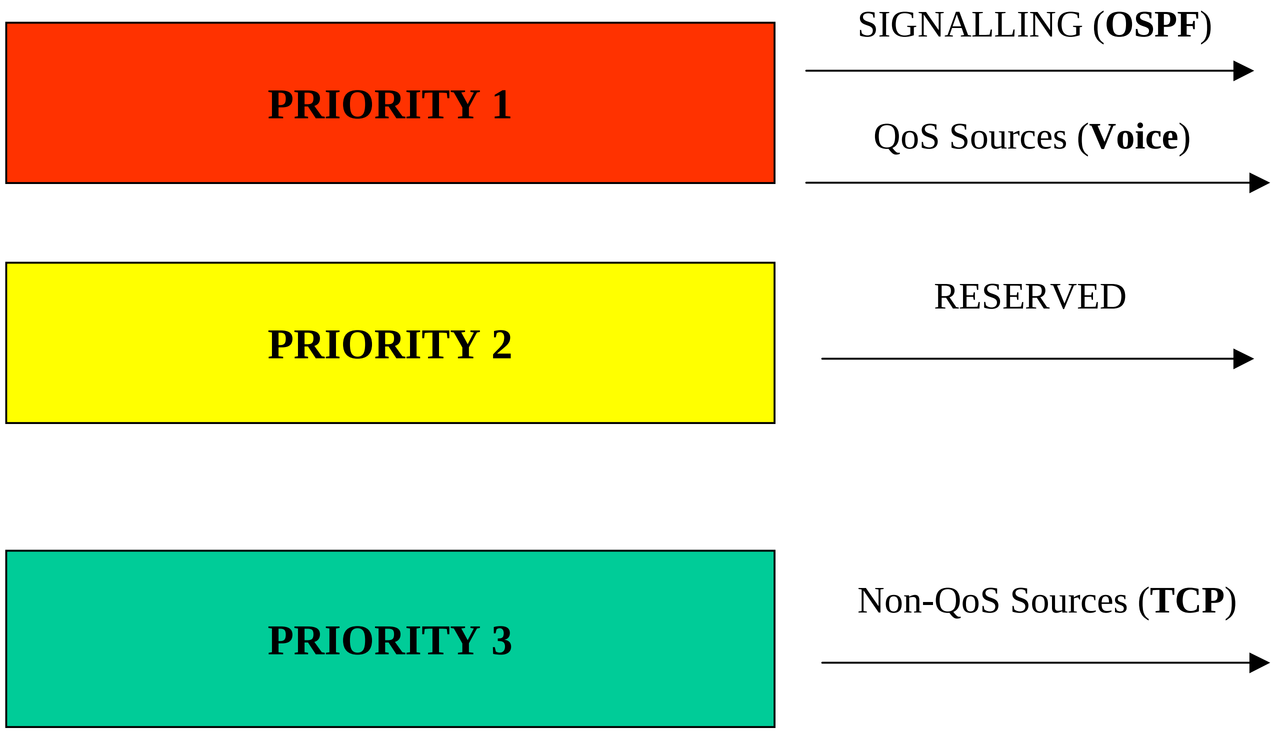
CAC and packet forwarding

- **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)

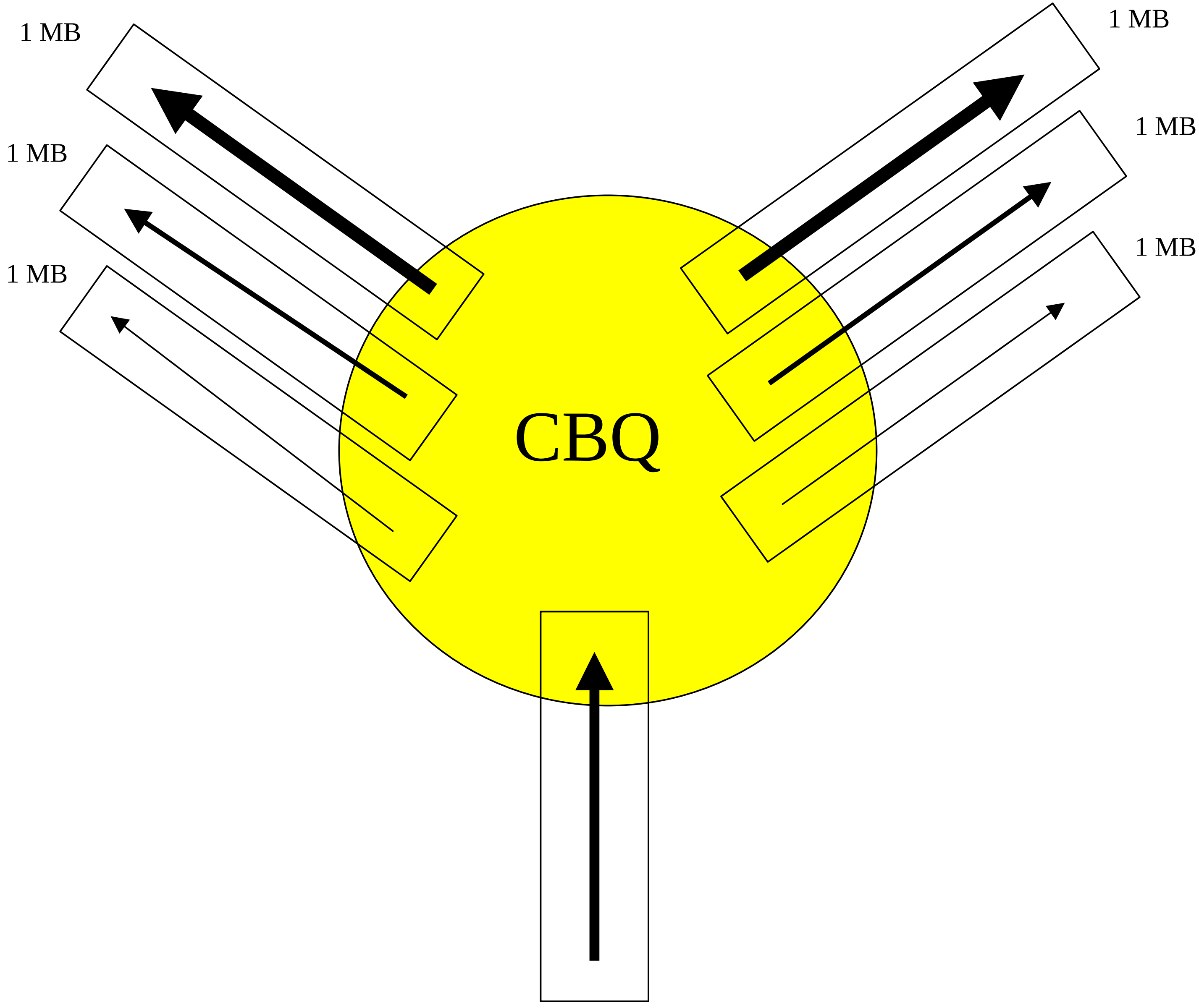
Application I: IP Telephony

- 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

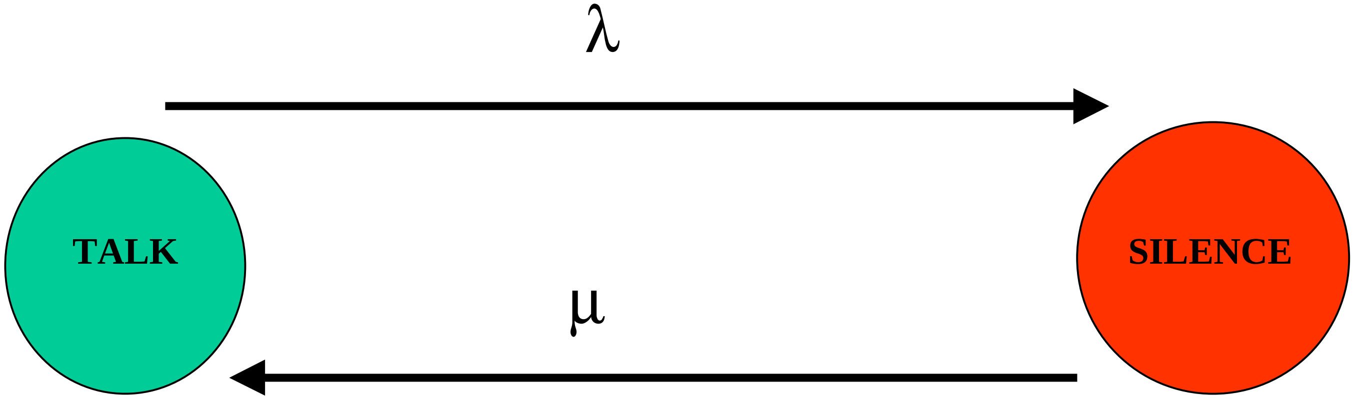
QoS simulator: priority queues



QoS Simulator: buffer allocations



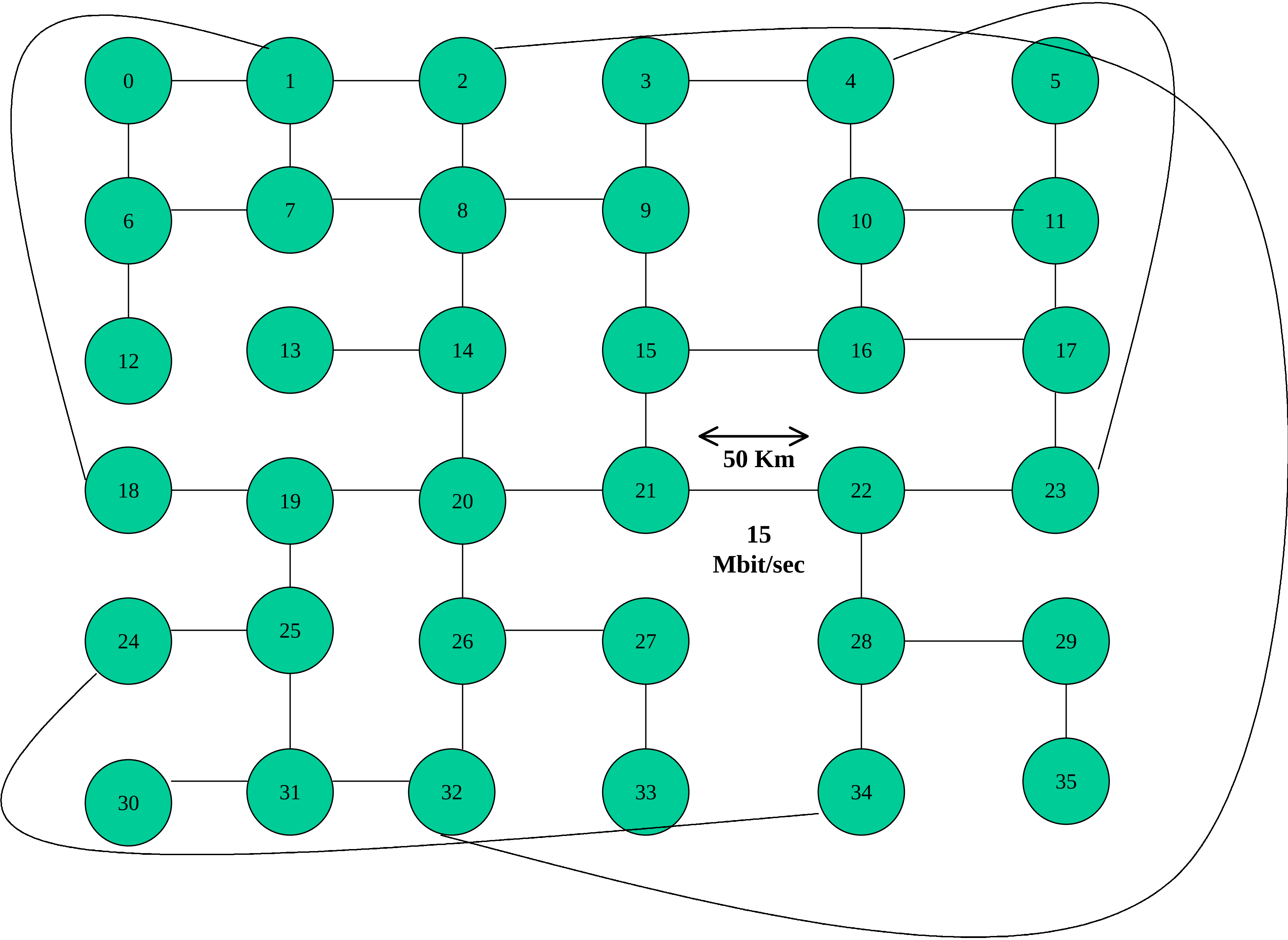
QoS Simulator: Voice Source Modeling



$1/\lambda = 352$ 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
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
50 Km
15
Mbit/sec

Simulation Parameters

- 10 Minute Simulation Runs
- Each voice connection lasts of 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

Light Load Experiment

Voice Call interarrival T = 375ms; No TCP traffic

	MC Bellman Ford	Minhop
#calls generated:	1120	1120
#calls accepted:	1120	1120
% packet lost:	0%	0%
% packet above Delay Thresh:	0%	0%
% packet below Delay Thresh:	100%	100%

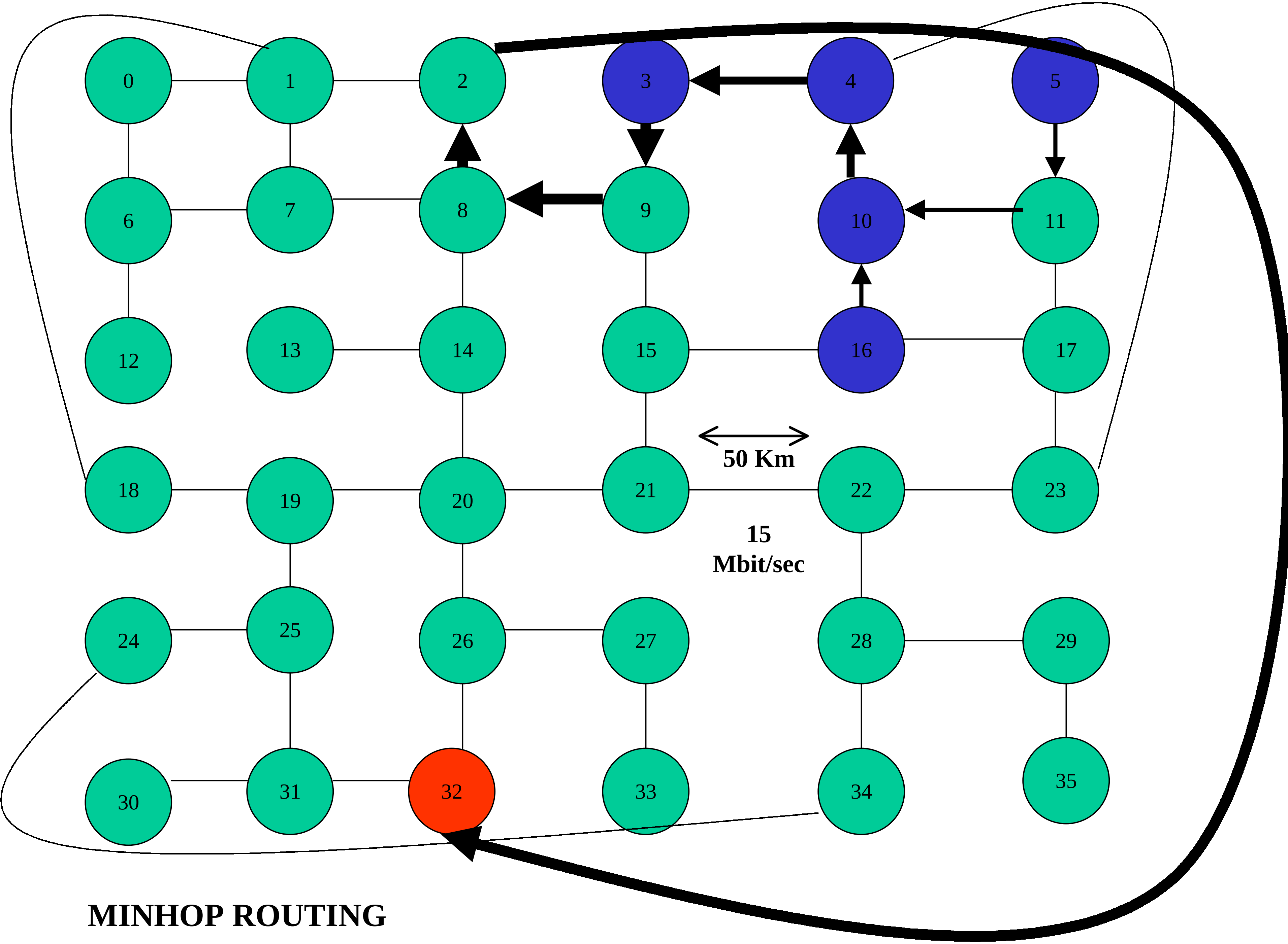
DELAY THRESHOLD = 100ms

Heavy Load Experiment

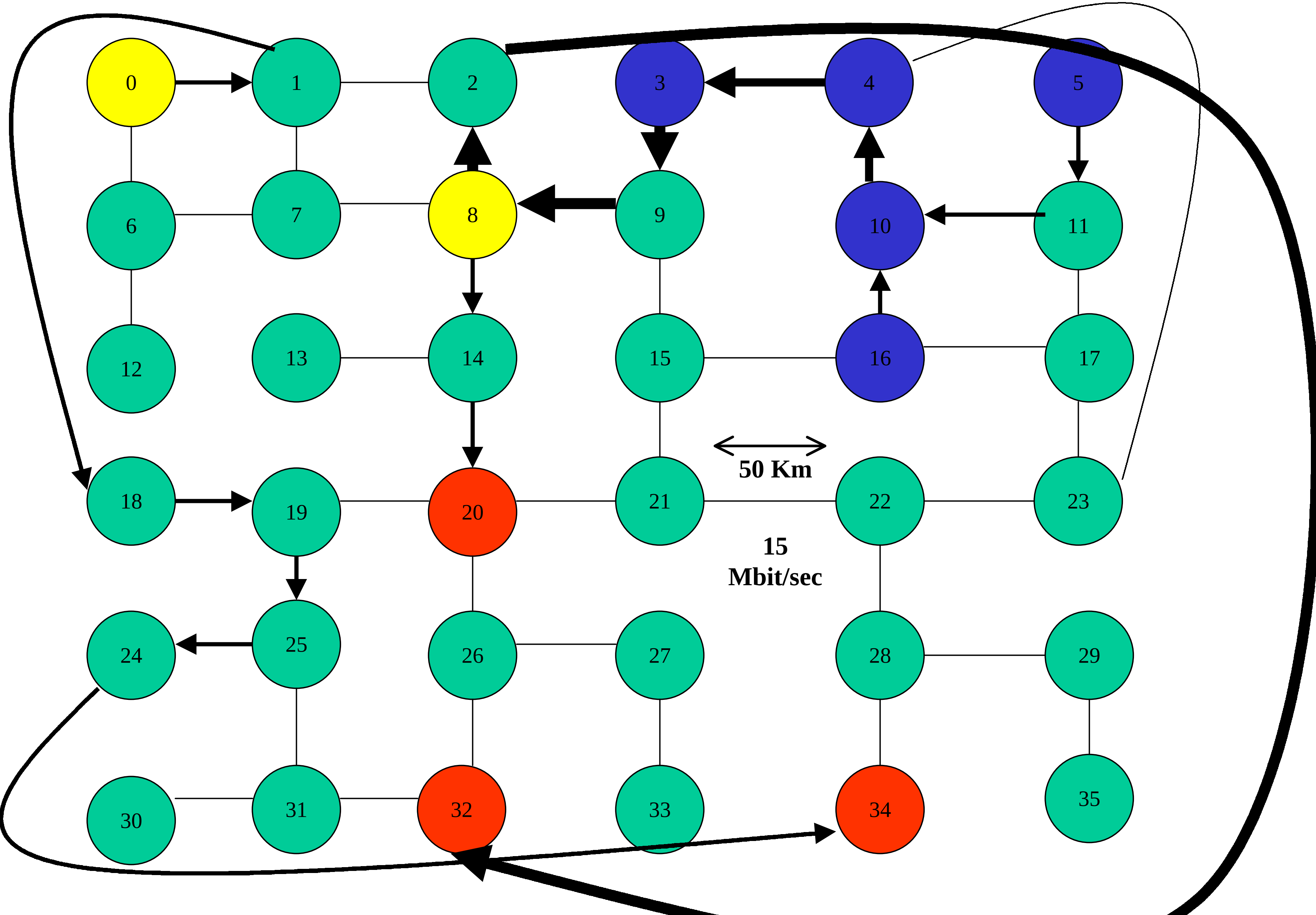
Voice Call interarrival T = 150ms; No TCP traffic

	MC Bellman Ford	Minhop
#calls generated:	2800	2800
#calls accepted:	2800	2800
% packet lost:	0.0%	22.61%
% packet above Delay Thresh:	0.0%	48.83%
% packet below Delay Thresh:	100.0%	28.56%

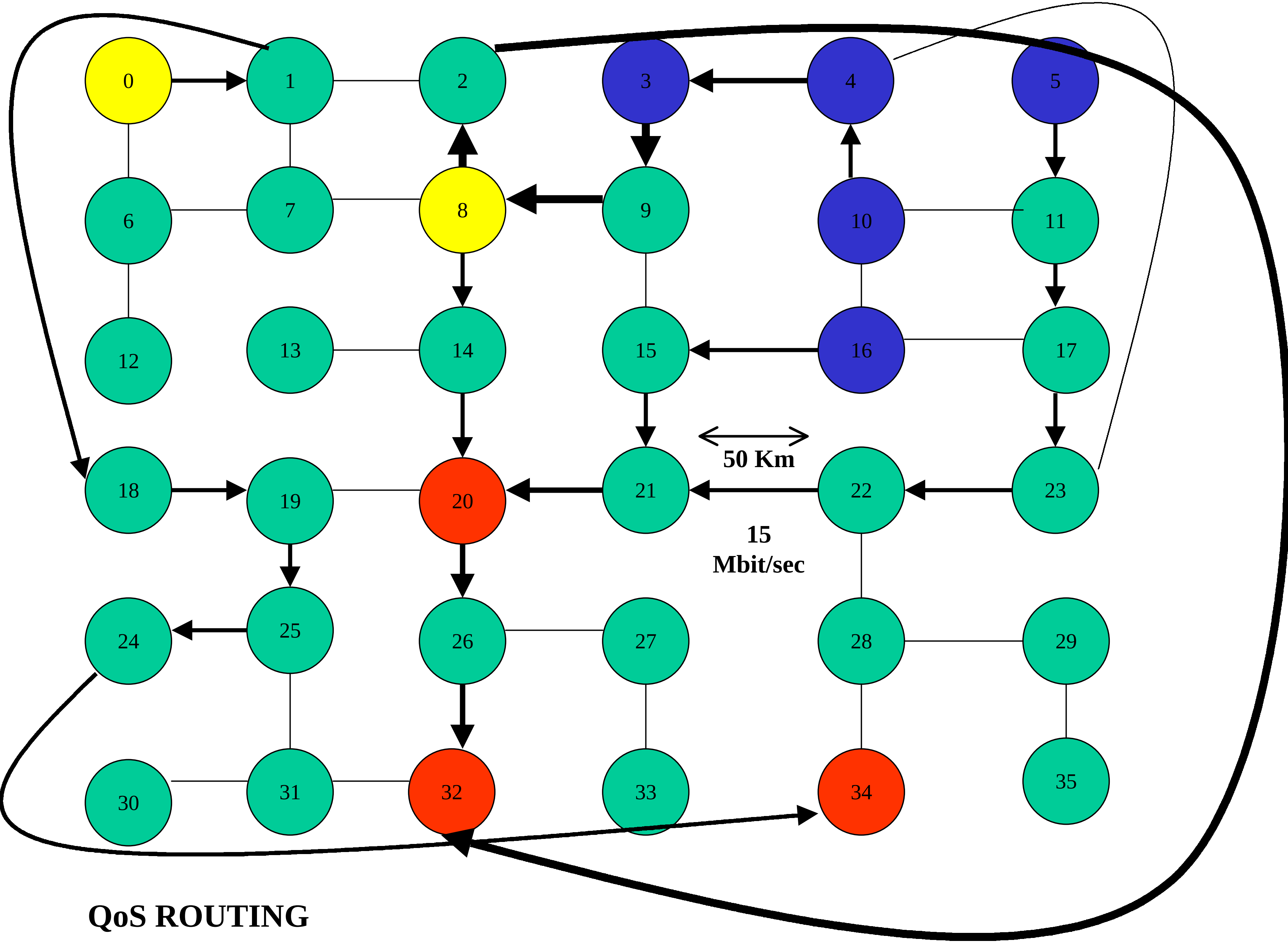
DELAY THRESHOLD = 100ms



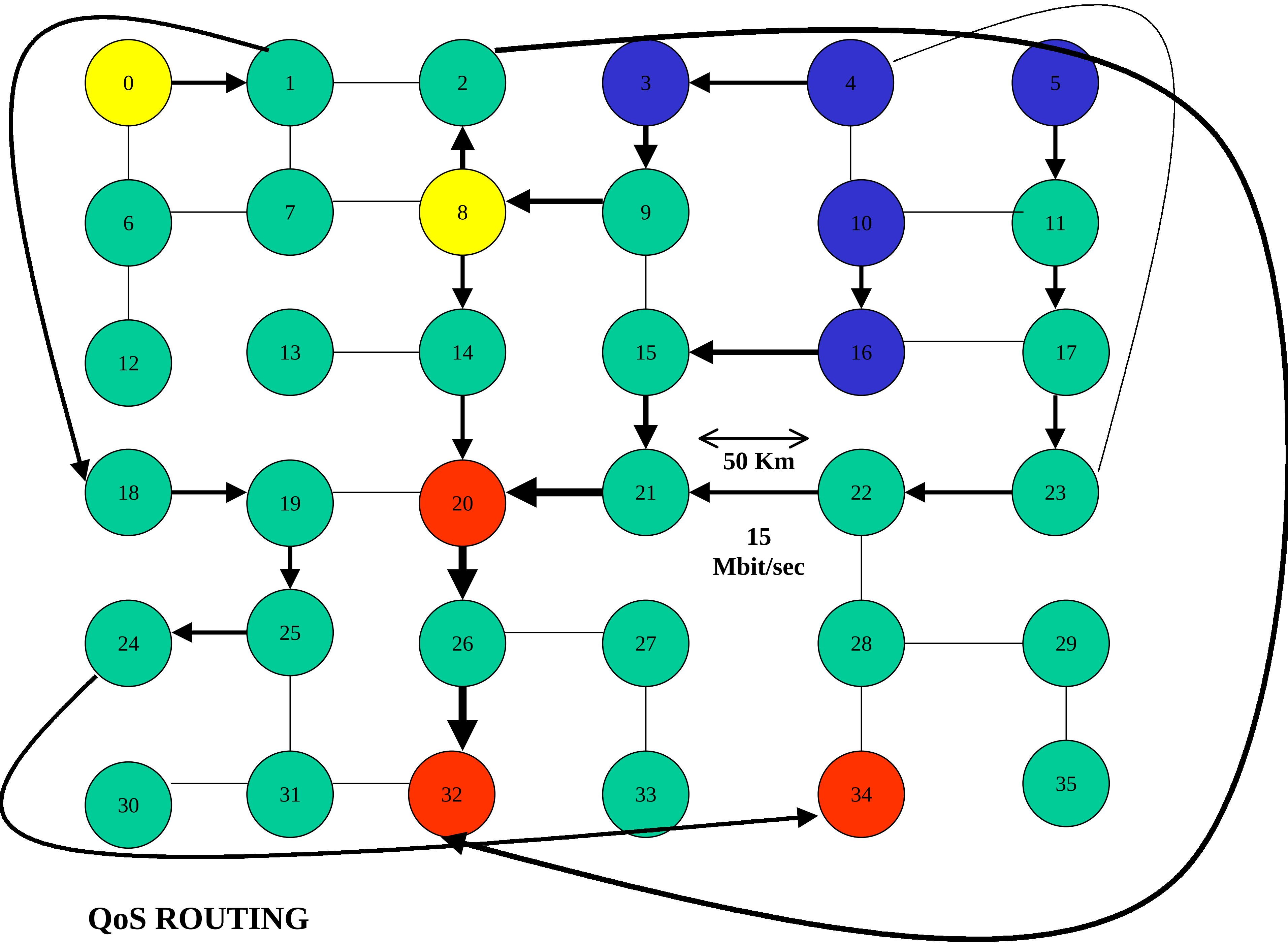
MINHOP ROUTING



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

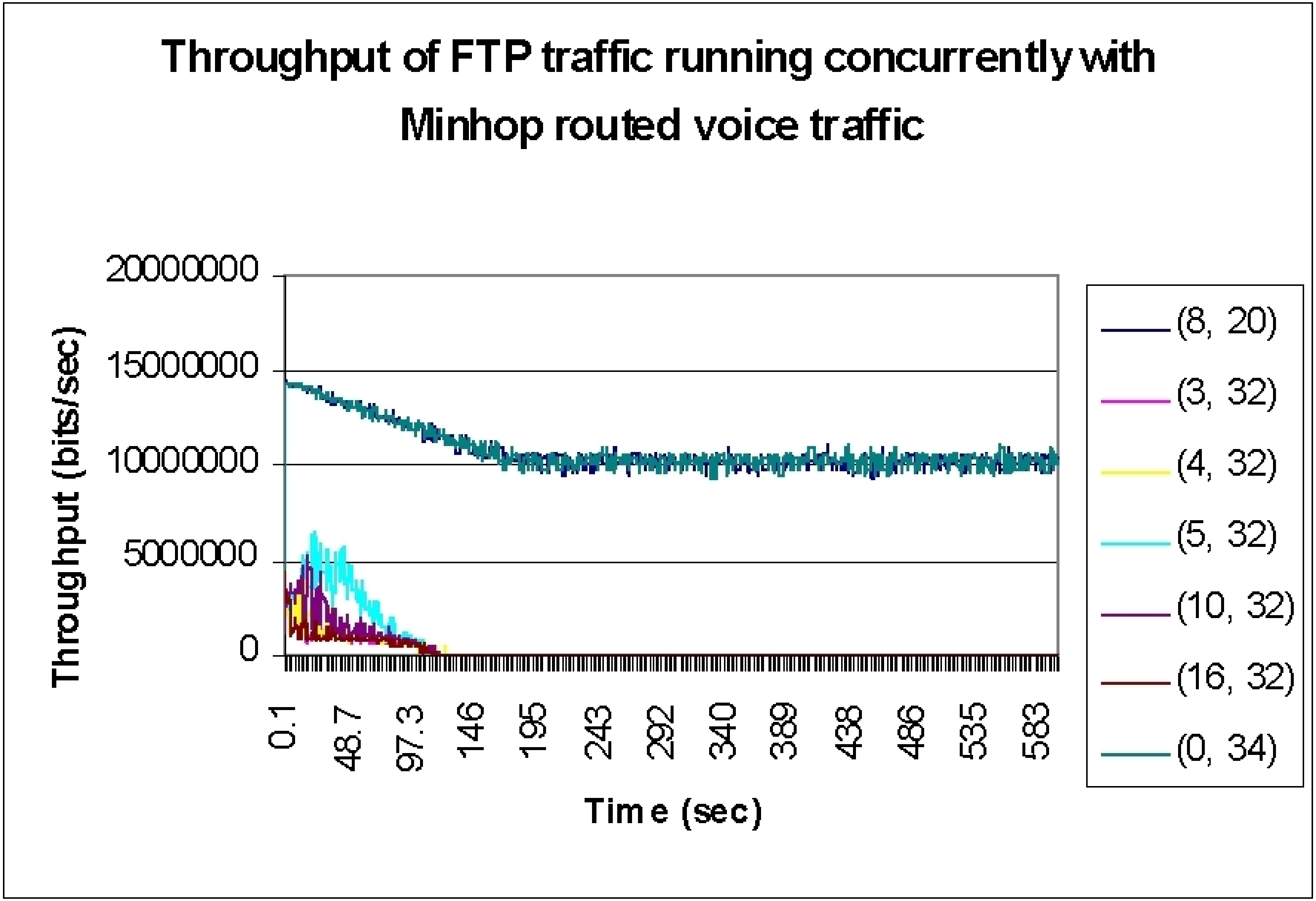
Heavy Load Experiment with TCP

Voice Call interarrival T = 150ms

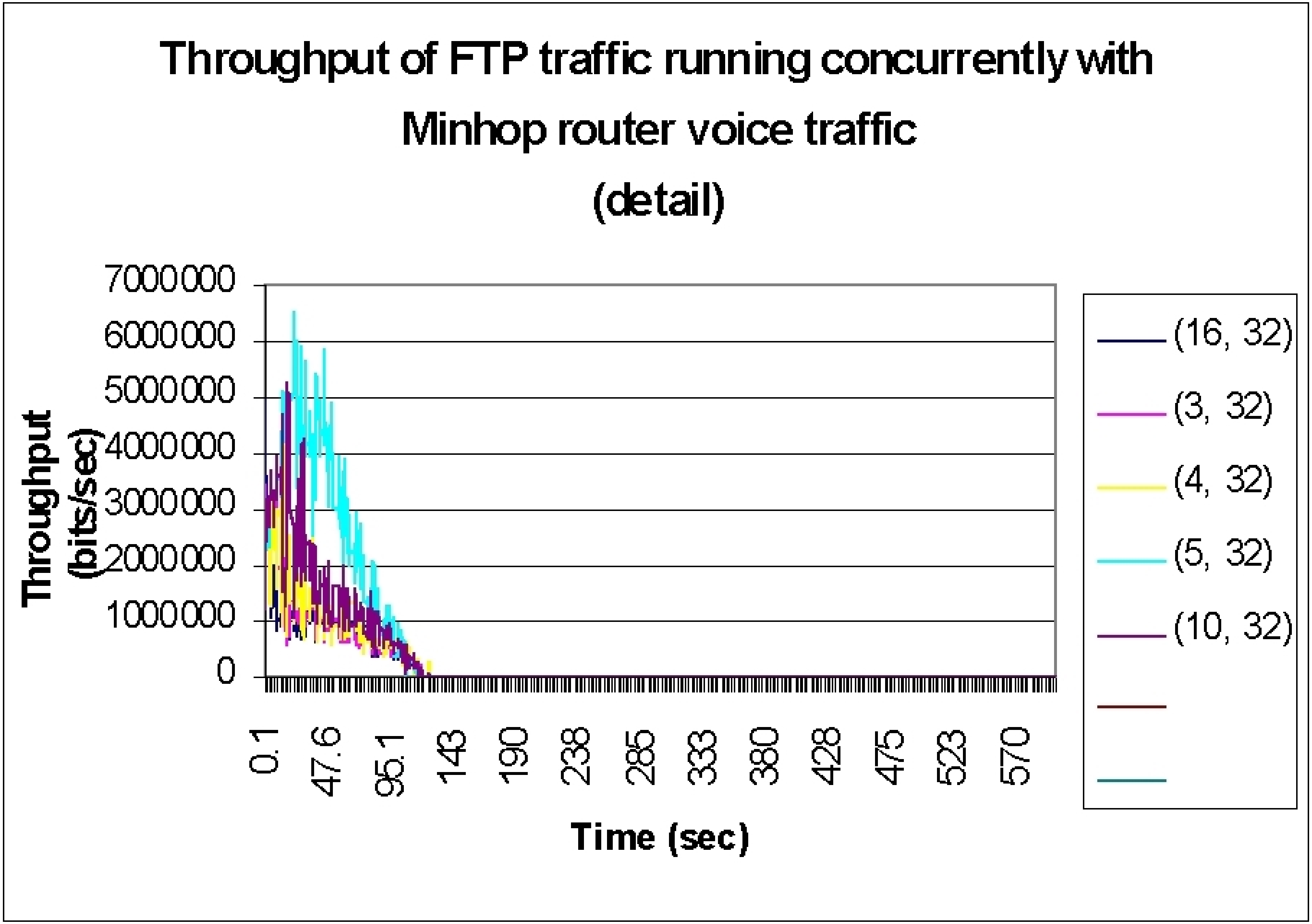
- TCP source/destination pairs same as for voice
- FTP infinite backlog
- TCP Reno

TCP steady state performance with QoS Routing

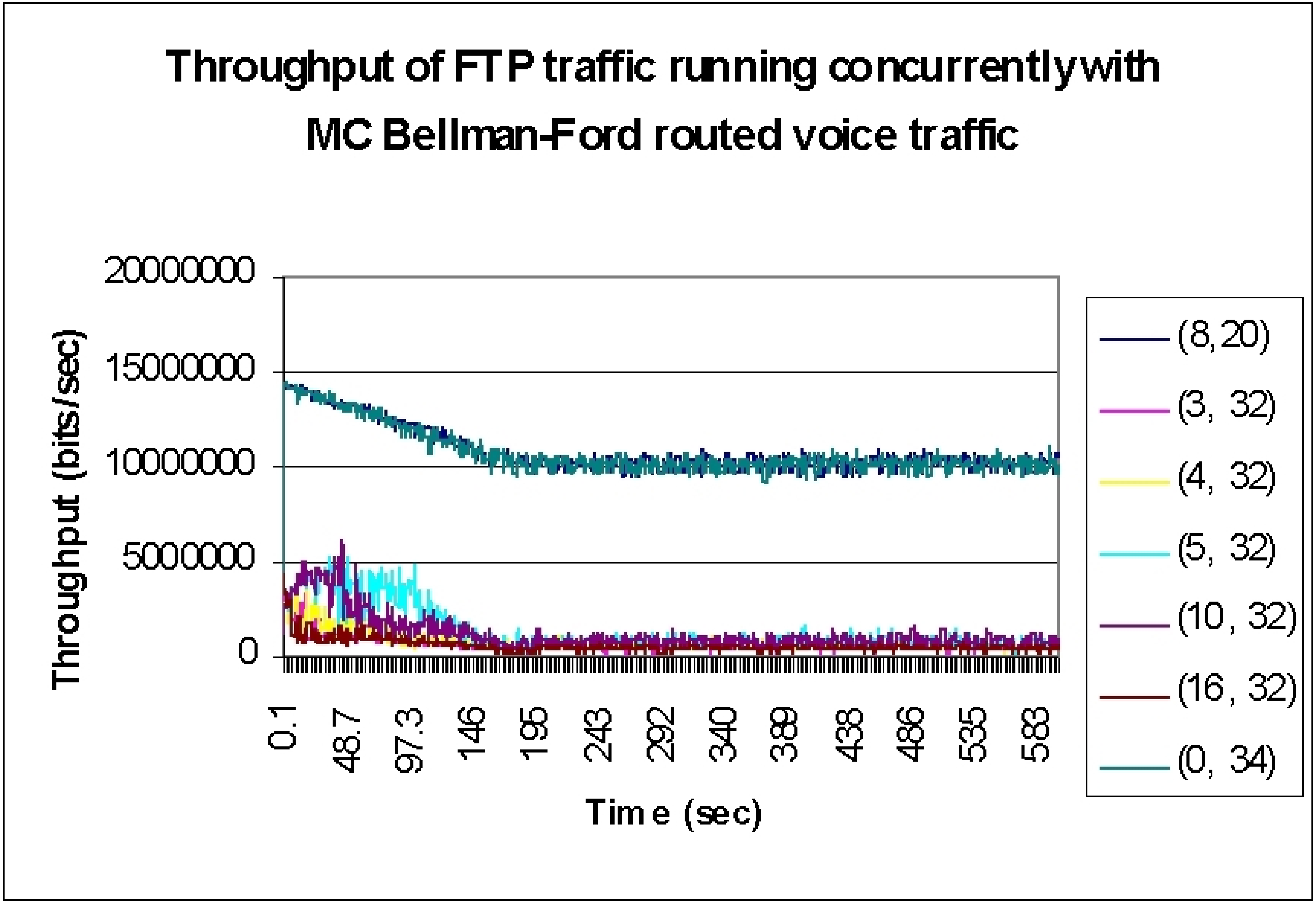
(src, dst)	Throughput
(3, 32)	471353bps = 471Kbps
(4, 32)	578806bps = 579Kbps
(5, 32)	729800bps = 730Kbps
(10, 32)	796863bps = 797Kbps
(16, 32)	404768bps = 405Kbps
Sum	2981590bps = 3.0Mbps

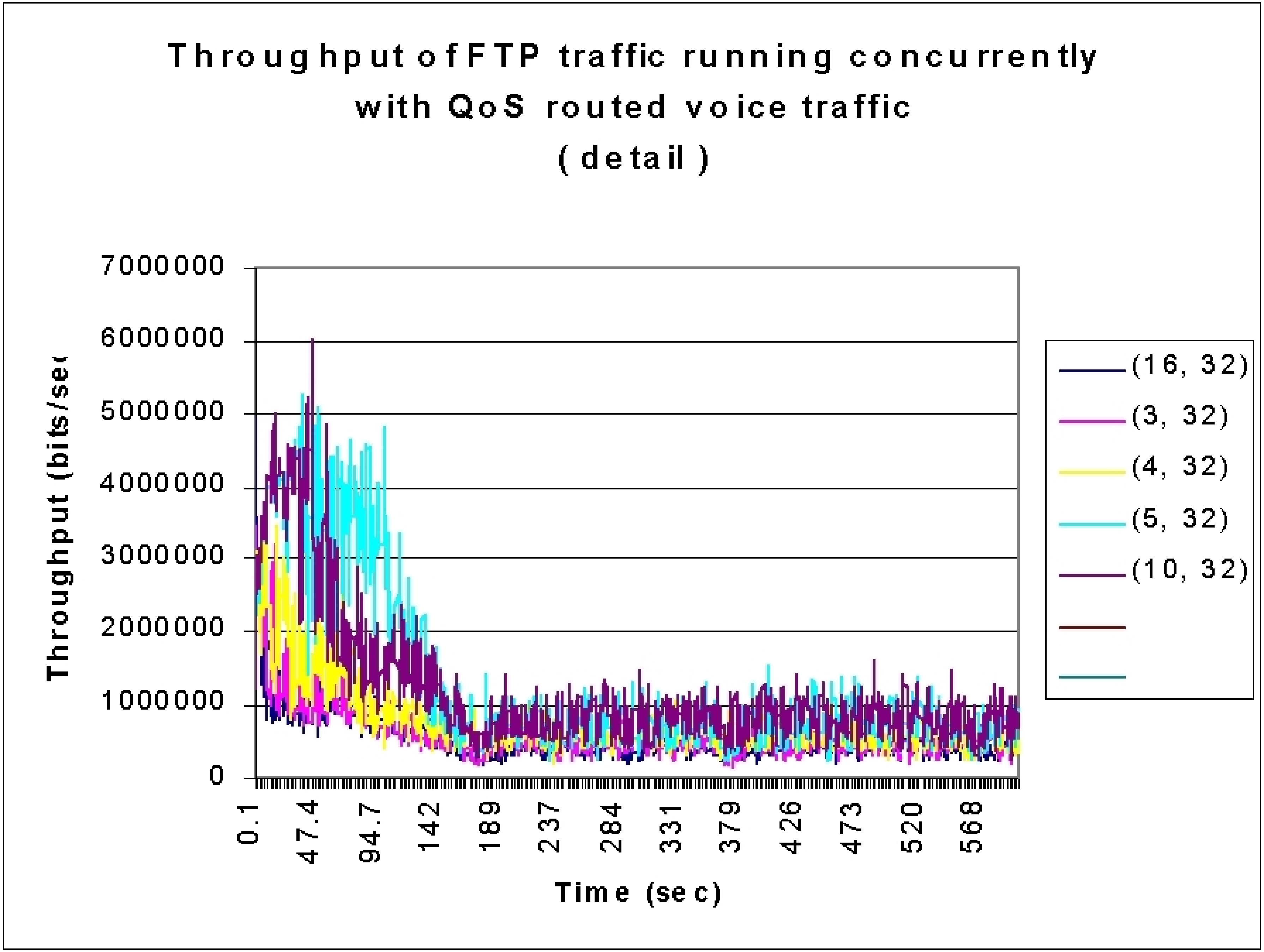


Throughput of FTP traffic running concurrently with Minhop routed voice traffic
20000000
15000000
10000000
5000000
0
Throughput (bits/sec)
0.1
48.7
97.3
146
195
243
292
340
389
438
486
535
583
Time (sec)
(8, 20)
(3, 32)
(4, 32)
(5, 32)
(10, 32)
(16, 32)
(0, 34)

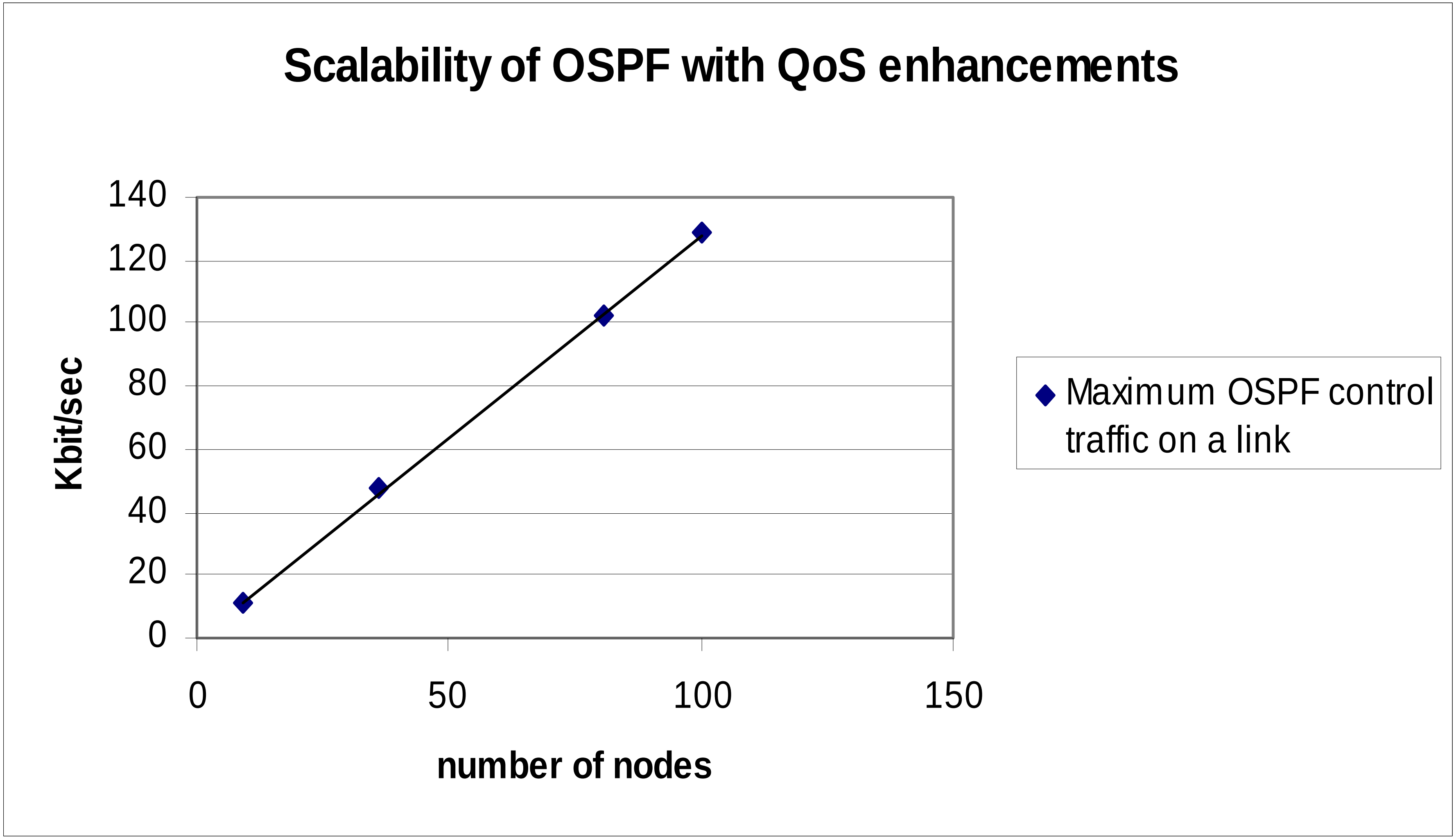


Throughput of FTP traffic running concurrently with Minhop router voice traffic (detail)
7000000
6000000
5000000
4000000
3000000
2000000
1000000
0
Throughput (bits/sec)
0.1
47.6
95.1
143
190
238
285
333
380
428
475
523
570
Time (sec)
(16, 32)
(3, 32)
(4, 32)
(5, 32)
(10, 32)

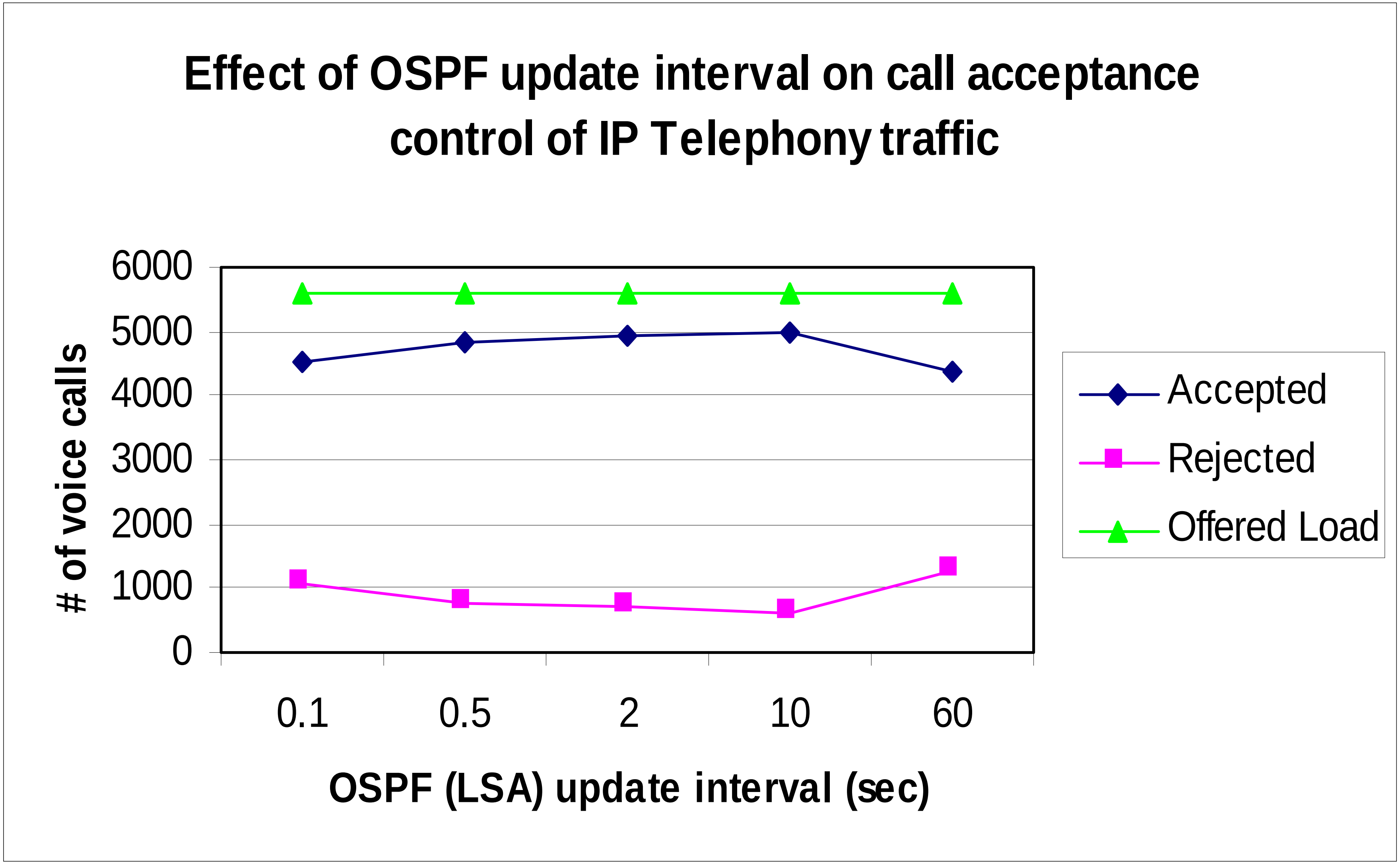

Throughput of FTP traffic running concurrentlywith
MC Bellman-Ford routed voice traffic
Throughput (bits/sec)
20000000
15000000
10000000
5000000
0
0.1
48.7
97.3
146
195
243
292
340
389
438
486
535
583
Tim e (sec)
(8,20)
(3, 32)
(4, 32)
(5, 32)
(10, 32)
(16, 32)
(0, 34)



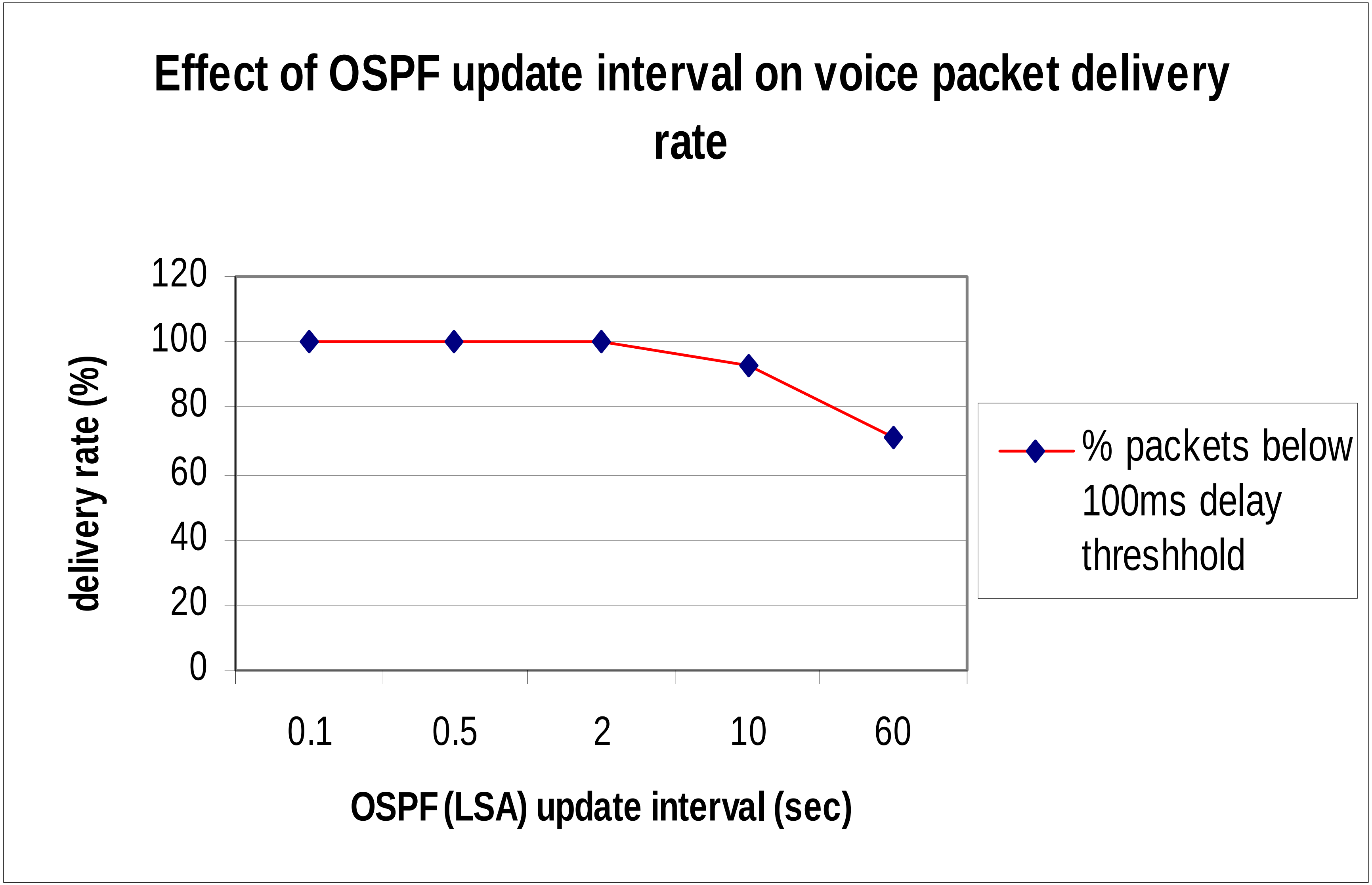
Throughput of FTP traffic running concurrently with QoS routed voice traffic (detail)
Throughput (bits/se
7000000
6000000
5000000
4000000
3000000
2000000
1000000
0
0.1
47.4
94.7
142
189
237
284
331
379
426
473
520
568
Time (sec)
(16, 32)
(3, 32)
(4, 32)
(5, 32)
(10, 32)



- **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

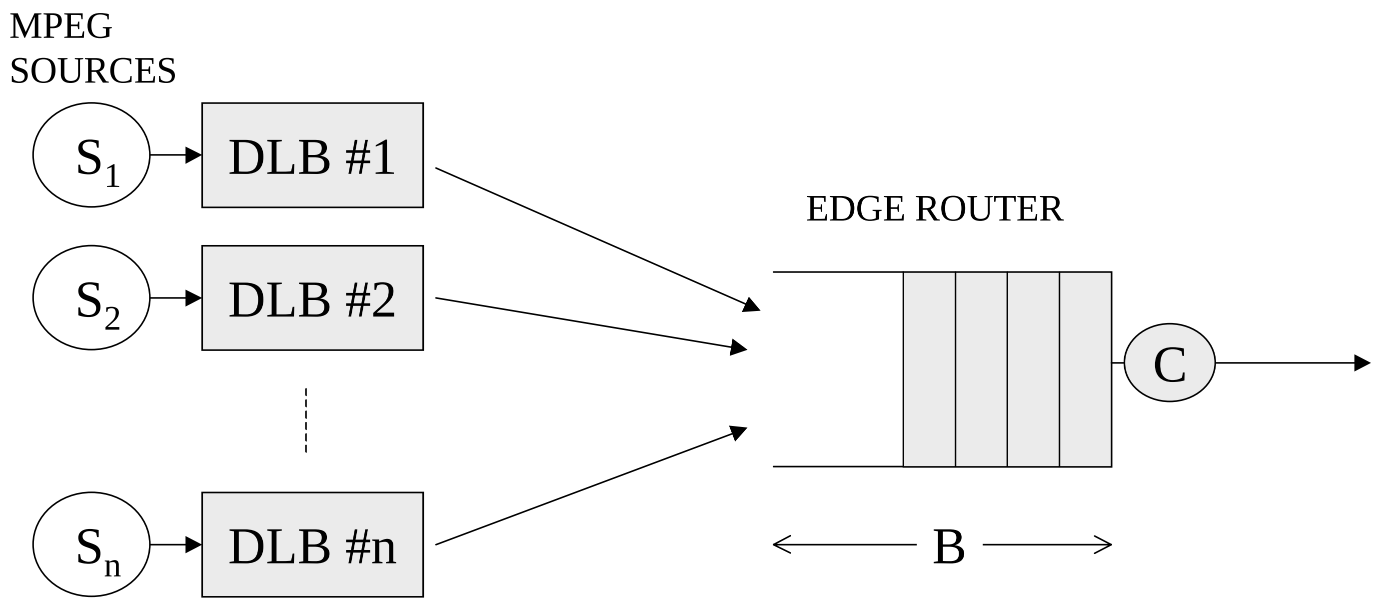


75ms voice call generation rate

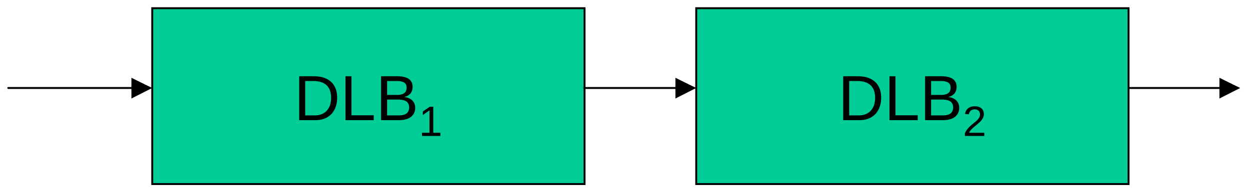
Application II: MPEG video

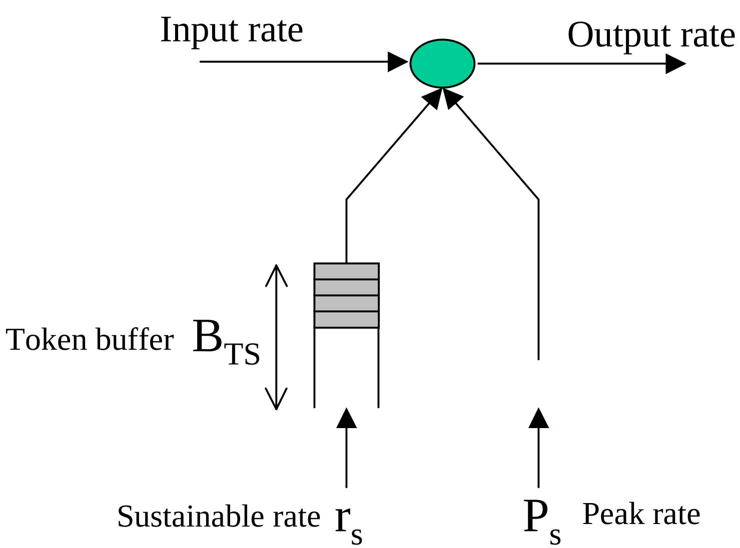
- **Measurement** based CAC (which worked well for IP telephony) is now replaced by **reservation** based CAC
- RSVP type signaling required
- Effective bandwidth & buffer reservations
- 36 node grid-type topology
- Trunk capacity = 5.5 Mbps
- Inputs = Measured MPEG traces
- QoS guarantees: no-loss; delay < Dmax
- Simulation platform: PARSEC wired network simulation

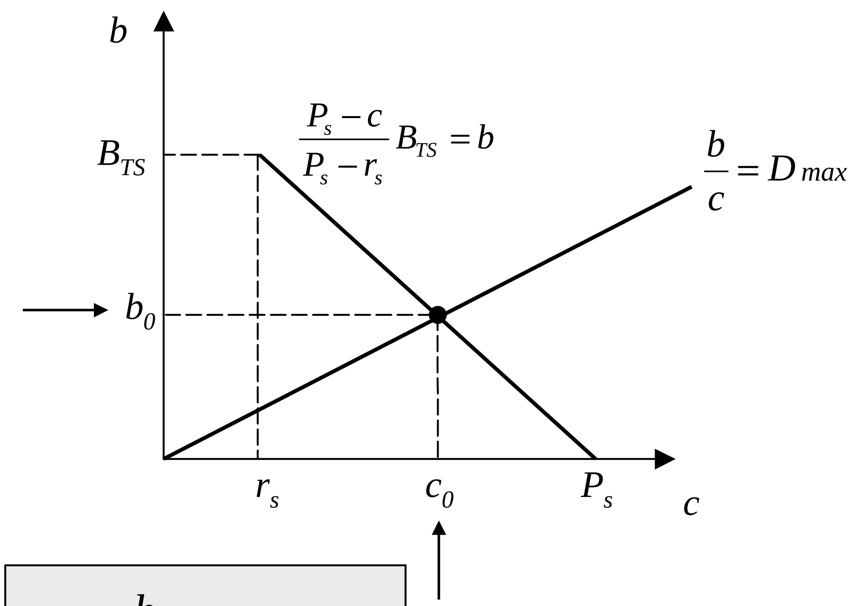
NETWORK ACCESS SECTION



DUAL LEAKY-BUCKET REGULATOR







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

LOSSLESS ALLOCATION

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

SCHEDULING

NETWORK CORE SECTION

- Each router keeps a resource bookkeeping table:

OUGOING LINK; AC ; NS_1 ***,*** NS_2 ***,...*** NS_n **;**

AC*:* Available Capacity

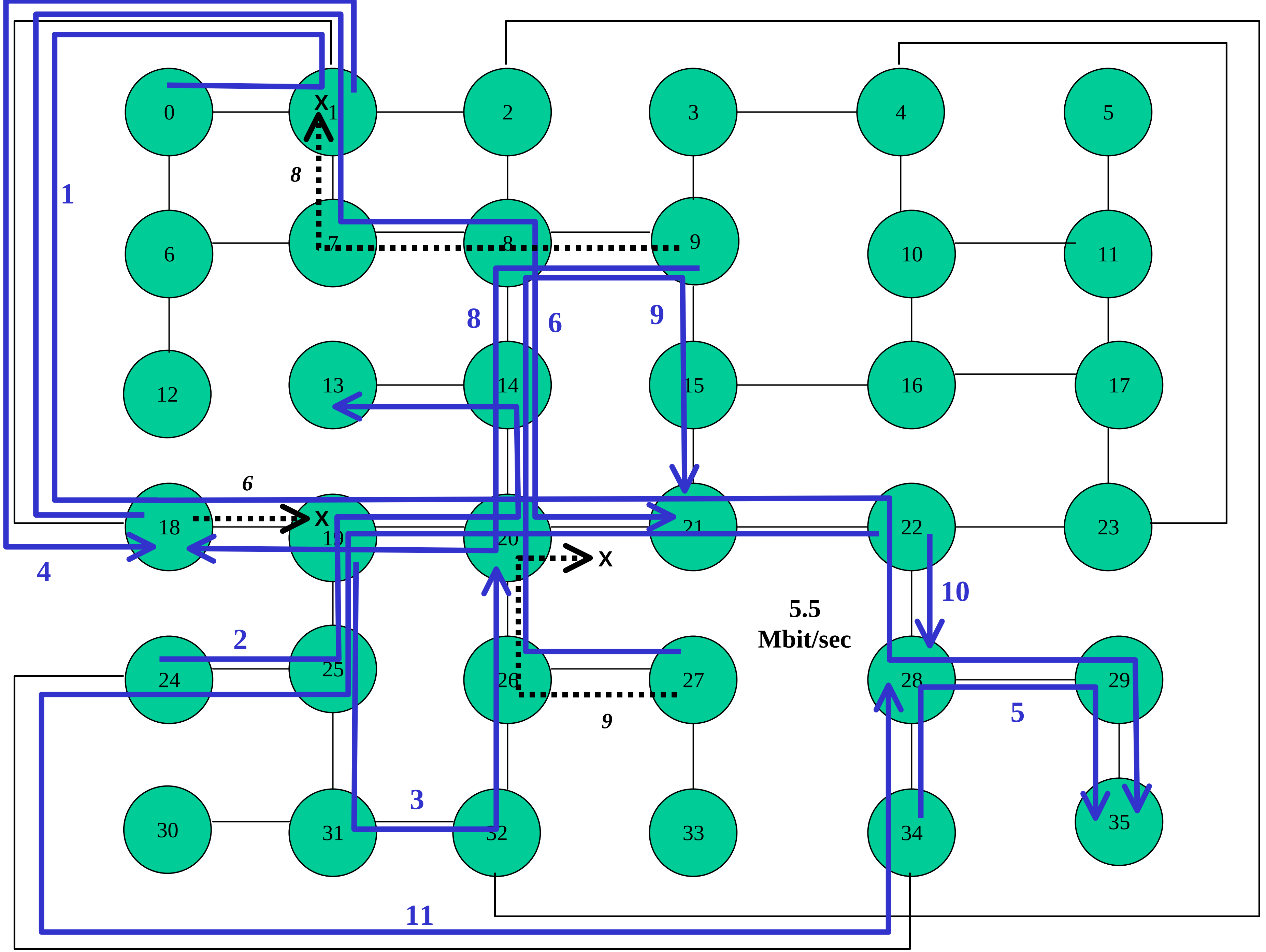
NS_k *, k=1,…,n*: the number **additional** flows belonging to class S_k that the router can accept within QoS constraints

- Each router loads the **AC** and **NS** information in the OSPF Link State Update packet

SIMULATION RESULTS

Bandwidth/link: 5.5 Mbps unidirectional
Dmax: 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	
Number of packets sent: 52866 - Number of packets lost: 0 Number of packets received: 52866 - Max delay for video packets 0.080648 s					


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
2
3
4
5
6
8
9
10
11
8
6
9
X
X
X
5.5
Mbit/sec

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

- Use of MPLS instead of source routing
- extension to hierarchical OSPF
- extension to Interdomain Routing
- extension to multiple classes of traffic
- Statistical allocation of MPEG sources