Numerical Evaluation of Hierarchical QoS Routing *

Sungjoon Ahn, Gayathri Chittiappa, A. Udaya Shankar
Computer Science Department and UMIACS
University of Maryland, College Park

CS-TR-3905
April 30, 1998

Abstract

We develop a numerical evaluation method for adaptive hierarchical QoS routing, and demonstrate its viability by application to two networks. Our approach models aggregation and delayed feedback in a straightforward way, and is scalable to the large networks needed to evaluate hierarchical routing.

1 Introduction

Packet-switched networks in the near future are expected to support applications with a wide variety of quality-of-service (QoS) requirements, such as bounded end-to-end delay. Routing and admission control play a key role in this respect. Routing provides an end-to-end path for the application. Admission control specifies the amount of resources (bandwidth, buffers, scheduling priority, etc.) needed on the path to achieve the desired QoS.

We can distinguish the following activities:

- *Router view maintanence*: Each router maintains a view of the network state that is periodically refreshed by routing updates. The view may be simple and relatively static (e.g. up/down status of links), or complex and dynamic (e.g. current load and types of flows on links).
- *Path selection*: When a connection request of some QoS arrives, the source router chooses a path based on its view and a "path selection rule".

*This work is supported partially by ARPA contract number DABT6396C0075 and DoD contract number MDA90497C3015 to the University of Maryland. It should not be interpreted as representing the opinions or views of ARPA, DoD, or the U.S. Government.

- *Path set up*: Once a path is chosen, a set up message travels along the path reserving resources at each hop according to the admission control rule. If successful, the connection is established, otherwise the connection is blocked (in which case, a retry may be made on another path).

Router views can be "flat" or "hierarchical". In flat routing, a router's view has uniform detail over the network, typically, a graph with a vertex for every router, an edge for every link, and a "qos" attribute for every vertex and edge indicating resource availability (e.g. available bandwidth).

Most large networks, including internets, use hierarchical routing. Here, routers are grouped hierarchically into "logical groups", and each group maintains a view that contains its subgroups, the peer groups in its parent group, and so on. An "aggregate" qos attribute is maintained for each group, obtained by applying an aggregation operation on the qos attributes of the subgroups. An example of a two-level area hierarchy is shown in Figure 1. Each router is a level-0 node, and groups of routers and their included links form the level-l nodes.

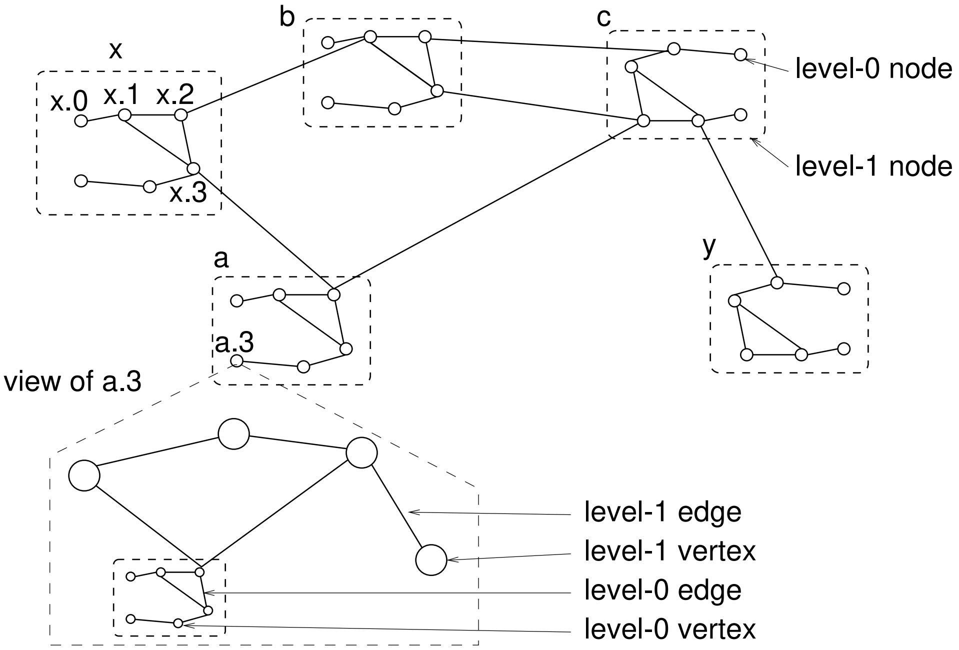

Figure 1: A Network with a 2-level Routing Hierarchy

Note that in flat routing, the source router of a connection request can supply a link-level path for admission control, whereas in hierarchical routing it can only supply a higher level path that is resolved to a link-level path during set up. For example, in Figure 1, $x.0$ may choose $x.0$, $x.1$, $x.2$, $x.3$, a, c, y, $y.1$ as a path to $y.1$, where $x.0$, $x.1$, $x.2$, $x.3$ and $y.1$ are level-0 nodes and a, c, and y are level-1 nodes. During setup, the node a is expanded by a router in a into a path going through one or more routers in a, and similarly with c and y.

Hierarchical routing has much less overhead than flat routing. But it has less accurate views, and this can result in increased blocking. Aggregate attributes that are overly optimistic make it more likely that selected paths will do not pass the admission control test during set-up. Overly pessimistic aggregate attributes make it more likely that the selection rule will not find a path with adequate resources even though such paths exist.

While admission control has been studied extensively, our understanding of the routing aspect is very limited, specifically, the effect of aggregation, update interval, selection rule, etc., on the likelihood that the chosen path has adequate resources to satisfy admission control.

This paper provides an approach to examine such issues. We consider networks with application workload, admission control, and hierarchical routing. We model these networks by time-dependent queues. and obtain the time evolution of ensemble performance metrics using a fast numerical approximatin technique, referred to as the *Z-iteration*. The approach is illustrated on networks with two-level hierarchical routing and effective bandwidth admission control. Evaluations are also donde for flat routing, thereby quantifying the effect of aggregation. We also validate the Z-iteration results against simulations (which are computationally much more expensive).

Unlike simulation. our numerical approach is scalable to the large networks needed to evaluate hierarchical routing. It also models aggregation and delayed feedback in a straightforward way, unlike analytical methods which are usually too "coarse".

The rest of the paper is organized as follows. Section 2 describes our network model. Section 3 describes the time-dependent multi-class multi-resource queue corresponding to a network. Section 4 describes the Z-iteration numerical solution method for time-dependent queues. Section 5 is an application to two example networks. Section 6 concludes with future work.

2 Network Model

The following subsections describe the kind of network we consider, specifically, the views maintained in hierarchical and flat routing, the application workload, the admission control rule, and the path selection rules.

2.1 Hierarchical Routing Views

Each router is a level-0 node, and groups of routers and their included links form the level-l nodes. A router's identifier has the form $x.i$, where x identifies the group and i identifies the node within the group. A level-1 node identifier consists of just the group id, e.g., x. A link between nodes $x.i$ and $y.j$ is denoted by $(x.i, y.j)$.

In the two-level hierarchical routing, the view of a router in group x is the graph consisting of the following components:

- Level-0 vertices $x.i$ for every level-0 node i in group x.
- Level-1 vertices y for every group y other than x.
- Level-0 edges $(x.i, x.j)$ for every link $(x.i, x.j)$ in group x.
- Level-1 edges $(x.i, y)$ for every link $(x.i, y.j)$ such that $x \neq y$.
- Level-1 edge (y, z) for every link $(y.i, z.j)$ such that $x \neq y \neq z$.

Each vertex and edge has a qos attribute reflecting available bandwidth. There is no delay attribute because the admission control rule does not need it.

- The qos attribute of a level-0 edge $(x.i, y.j)$ is the available bandwidth of the corresponding link, and is denoted by $(x.i, y.j).bw$.
- The qos attribute for a level-0 vertex is implicitly set to infinity.
- The qos attribute of a level-1 vertex y, denoted $y.bw$, is defined by the following aggregation:

  $$y.bw = \frac{1}{\mid L \mid} \sum_{\forall l \in L} l.bw$$

  where L is the set of all links internal to group y. Intuitively, $y.bw$ indicates the average available bandwidth connectivity between routers in y.
- The qos attribute of a level-1 edge $(x.i, y)$, denoted $(x.i, y).bw$, is

  $$(x.i, y).bw = \frac{1}{\mid L \mid} \sum_{\forall p \in L} p.bw$$

  where L is the set of all links between $x.i$ and any node of y.
- The qos attribute of a level-1 link (y, z), denoted $(y, z).bw$, is

  $$(y, z).bw = \frac{1}{\mid L \mid} \sum_{\forall p \in L} p.bw$$

  where L is the set of all links between group y and group z.

In a routing protocol implementation, the above aggregation would be computed and disseminated by the special "leader" routers. In our evaluation, we assume that dissemniated information reaches other nodes instantaneously. This is justifiable because the time to propagate routing information is negligable compared to the routing update period.

2.2 Flat Routing Views

In flat routing, the view of a router $x.i$ is the graph consisting of vertices for every router in the network, edges for every link in the network. Each edge has a qos attribute equal to the available bandwidth of the corresponding link,

2.3 Application Workload

The application workload is defined in terms of *user classes*. A user class represents a stream of connection requests with the same source and destination nodes and the same traffic and QoS parameters. Specifically, user class i has the following attributes:

- source and destination nodes.
- Arrival rate of connection requests, denoted λ_i. The requests arrive as a time-dependent Poisson process.
- Average lifetime of a connection, i.e., from its establishment to its termination, denoted $1/\mu_i(t)$. The lifetime can have a time-dependent general distribution.
- QoS requirement can be arbitrary. Here we consider statistical delay bound $(D_i,\ \epsilon_i)$, meaning that Prob[end-to-end packet delay $> D_i] < \epsilon_i$ for any packet.
- Traffic descriptor can be arbitrary. Here we consider on-off traffic descriptor $(m_i, M_i m b_i)$, which means an on-off source with exponentially-distributed busy period of average duration b_i, mean transmission rate m_i, and peak transmission rate M_i.

2.4 Admission Control

The admission control policy is based on the concepts of effective capacity [1] and equal allocation [3].

Given a connection with on-off traffic descriptor (m, M, b) and statistical delay bound QoS (D, ϵ), the effective capacity EC and buffer requirement X of the connection can be computed as the fixed pont of

$$EC = M \cdot \frac{\beta - X + \sqrt{[\beta - X]^2 + 4X\rho\beta}}{2\beta}$$

where

- $\rho = m/M$ is the probability that the connection is active (on).
- $\beta = ln(1/\epsilon) \cdot b \cdot (1 - \rho) \cdot M$
- $X = D \times EC$ is the buffer space required by the connection.

Given a connection request and a path, the connection's end-to-end QoS is divided equally among the hops of the path. For a end-to-end statistical delay bound (D, ϵ) and a path of h links, this results in a "per-hop" QoS of $(\frac{D}{h}, \frac{\epsilon}{h})$.

The connection request setup succeeds at a hop iff the available bandwidth at the hop exceeds the effective capacity needed for the per-hop QoS. The available bandwidth is the link capacity minus the sum of the effective capacities of all the connections already using the link.

We are assuming the following: available buffer space at each router exceeds the needed amount (given by X); the only cause for delay is queuing at links; set-up and tear-down of a connection happens at every hop on the route instantaneously.

If the chosen path is unable to satisfy the QoS requirements of the requesting application, the connection is blocked. There is no attempt to find analternate route.

2.5 Path Selection in Flat Routing

For each user-class, we restrict the set of possible paths to minimum-hop and next-to-minimum-hop paths. This is acceptable because using a longer path for a connection ties up resources at more intermediate nodes, thereby decreasing network throughput. Note that there may be more than one minimum-hop or next-to-minimum-hop path.

Each path R is assigned a weight $W_R(t)$ indicating the fraction of connection requests of the user-class that will be routed on R at time t. The weighting function $W_R(t)$ is, in general, a function of the router's view, to be chosen to achieve a high likelihood of successful setup. In this report we use weights defined by [2]:

$$W_R(t) = \frac{F_R(t)}{H_R(t) \cdot L_R(t)}$$

where $H_R(t)$ is the path hop length, $L_R(t)$ is the path utilization (average of the utilizations of the path's links), and $F_R(t)$ is a measure of the feasibility of the path. Here, $F_R(t)$ equals 1 iff the path (as stored in the router's view) satisfies the admission control rule, otherwise it equals 0.

2.6 Path Selection in Hierarchical Routing

This is like selection in flat routing except that

- The chosen path is a level-1 path (as found in the source router's view) rather than a level-0 path.
- The admission control rule is applied to level-0 links, and level-1 nodes, and level-1 links on the path.

- When a level-1 node in the path is expanded into a level-0 "sub-path" within the level-1 node, only level-0 sub-paths of minimum-hop and next-to-minimum-hop paths are considered. Thus two level-0 paths obtained from the level-1 paths of a user-class may differ in hop length by more than one.

3 Time-Dependent MCMR Queue Model

The above network model can be represented by a multi-class multi-resource self-service queue with time-dependent arrival rates. Each link in the network corresponds to a distinct resource, characterized by the link's capacity and available bandwidth. Each pair of user class and possible path corresponds to a distinct class of customers, referred to as a *routing class*.

The traffic of a routing class R of user class T at a link can be described by the following parameters:

- Instantaneous arrival rate $\lambda_R(t) = \lambda_T(t) \times W_R(t)$, where $W_R(t)$ is the (time-dependent) probability of selecting R's path. Note that $\lambda_R(t)$ can be time varying even if $\lambda_T(t)$ is not.
- Instantaneous service rate $\mu_R(t)$ is equal to $\mu_T(t)$
- QoS statistical delay bound $(\frac{D_T}{h}, \frac{\epsilon_T}{h})$, where h is the length of the path.
- On-off traffic descriptor equal to the traffic descriptor (m_T, M_T, b_T) for T.

4 Z-iteration

The Z-iteration is an efficient numerical method to compute accurate approximations of instantaneous ensemble metrics of time-dependant queuing systems. It is based on functional approximations of relationships between instantaneous metrics by the corresponding steady-state relationships. It also uses the decomposition approximation, with approximates multi-class multi-resource models by a collection of loosly-coupled multi-class single-resource models.

We summarize the method here for multi-class multi-resource self-service queues.

Consider a particular routing-class R at a link j. Let $N_{R,j}(t)$ be the average number of connections of routing-class R at link j. Let $B_R(t)$ be the blocking probability for routing class R, i.e., the probability that a connection request of R is blocked because of lack of resources at any one of the links on its path.

From the Chapmann-Kolmogorov equations of the MCMR queue, we can obtain the following flow equation for $N_{R,j}$:

$$\frac{dN_{R,j}(t)}{dt} = \lambda_R(t)[1 - B_R(t)] - \mu_R(t)N_{R,j}(t)$$

From the decomposition approximation, assuming independence between the available bandwidths at the different links of the path, we approximate $B_R(t)$ as follows:

$$1 - B_R(t) \approx \prod_{l \text{ in R path}} [1 - B_{R,l}(t)]$$

Now we assume a function $F(\cdot)$ that expresses $B_{R,l}(t)$ in terms of $N_{R,l}(t)$, that is

$$block_{R,l}(t) = F[N_{R,l}(t)]$$

We approximate the function $F(\cdot)$ by a function $G(\cdot)$ that expresses steady-state $B_{R,l}$ in terms of steady-state $N_{R,l}$, that is,

$$B_{R,l}(ss) = G[N_{R,l}(ss)]$$

Substituting this back, we get the following differential equation

$$\frac{dN_{R,j}(t)}{dt} = \lambda_R(t)[\prod_{l \text{ in R path}} (1 - G[N_{R,l}(t)])] - \mu_R(t)N_{R,j}(t)$$

$G(\cdot)$ can be obtained as

$$B = \sum_{i=j.cap-r.req+1}^{j.cap} P(i)$$

where $P(i)$ be the probability that link j has allocated bandwidth i, $j.cap$ is the capacity of link j, and $r.req$ is the bandwidth required by routing-class r on link j (which is the same for every link in its path).

The $P(i)$'s can be obtained as the fixed point of

$$i \cdot P(i) = \sum_{x \in X} \frac{\lambda_R}{\mu} \cdot x.req \cdot P(i - x.req) \quad i = 1, ..., j.cap$$

and

$$\sum_{i=0}^{j.max} P(i) = 1$$

where X is the set of routing-classes that require link j and $1/\mu$ is the average service lifetime. The λ_R/μ serves as an intermediate quantity in this fixed-point iteration.

5 Example Evaluations

We apply the evaluation method developed here for two specific networks. For each network, we compare instantaneous ensemble metrics for hierarchical and flat routing, for a

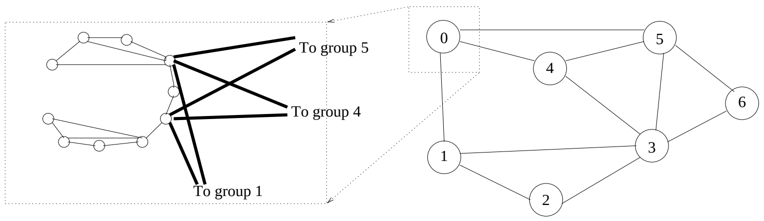

Figure 2: Topology 1 : 70 nodes, 111 links

Traffic type	Weight	$(\lambda_T,\, \mu,\, M,\, m,\, b,\, D,\, \epsilon)$
1	0.28	(0.3, 5.0, 60, 20, 0.1, 0.05, 0.00001)
2	0.44	(2.0, 0.5, 30, 10, 0.1, 0.05, 0.00001)
3	0.06	(2.0, 1.0, 30, 20, 0.1, 0.05, 0.00001)
4	0.22	(1.8, 0.5, 30, 10, 0.1, 0.05, 0.00001)

Table 1: Traffic types for Topology 1

variety of link speeds. We also validate the Z-iteration results by simulations, which were computationally more expensive by orders of magnitude.

The networks evaluated are shown in Figures 2 and Fig 3. Each network has several identical LANs connected by a backbone. Each level-1 node contains a LAN. All backbone links have capacity C_{ext}. All LAN links have capacity C_{int}. We vary the capacities of the LAN and backbone links The routing update period is 5 seconds.

Each network has an application workload of 50 user classes. to change the external load to the network. The user classes are distinguished by "traffic types" as shown in Table 1 and Table 2. A traffic type defines QoS and traffic descriptors. A traffic type, along with source and destination nodes, forms a user-class. The weight of a traffic type indicates the fraction of user classes of that type.

We present graphs showing the time evolution of the average number of connections in

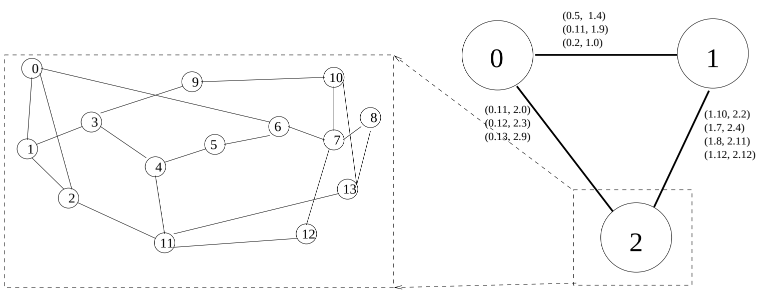

Figure 3: Topology 2: 42 nodes, 73 links

Traffic type	Weight	(λ_T, μ, M, m, b, D, ϵ)
1	0.20	(2, 1, 30, 20, .1, .05, .0001)
2	0.20	(2, .5, 30, 10, .1, .05, .0001)
3	0.20	(1.8, .5, 30, 10, .1, .05, .0001)
4	0.20	(.3, 5, 60, 20, .1, .05, .0001)
5	0.20	(.6, 2, 40, 20, 0.1, 0.05, 0.0001)

Table 2: Traffic types for Topology 2

the network, the blocking probability, the throughput (arrival rates of unblocked connections), and the average path length. The notation Hier(C_{ext}, C_{int}) indicates hierarchical routing with backbone link bandwidth C_{ext} and LAN link bandwidth C_{int}. Flat(C_{ext}, C_{int}) is the same but for flat routing. In all the graphs, abrupt changes occur at 5 second intervals, corresponding to the routing update instants.

Looking at the graphs displaying average path lengths we see that the average path length for hierarchical routing is more than for flat routing. This is as expected because in flat routing a router has more precise information about the state of the network, whereas in the hierarchical scheme a router has only an estimate of the state of each group. We also see that there is slightly more variation with respect to different external loads in the hierarchical scheme. This is because the lengths of paths for a given traffic-class differ by more than one in the hierarchical scheme. Thus the variation in the lengths of paths chosen is larger. However in flat routing, all the paths obtained for a traffic-class differ in path length by at most one.

Looking at the performance of both topologies in Fig 5 and Fig 6 we see that, in all the graphs (a), (b), and (c), the performance of hierarchical routing is very close to that of flat routing. The difference is less noticable in Topology 2 than in Topology 1. This could be because the difference in average path lengths in Topology 2 is slightly less than in Topology 1 causing a smaller difference in utilization of the network. In Topology 1 we see that for the case when C_{ext} is 600 and C_{int} is 200, the difference in measured statistics of flat and hierarchical is the largest. However this is not the case for Topology 2. In another set of experiments, the results of which are not presented here, the link capacities are of Topology 1 are varied such that the ratio 1:3 (200:600) is maintained. We find that the difference between flat and hierarchical for this ratio is more than when the link capacities are (100,500) or (600,600).

Overall, the graphs show that

1. The performance of the hierarchical scheme is very close to that of the flat scheme. Often the difference is negligible.

2. The average path length in the hierarchical scheme is higher.

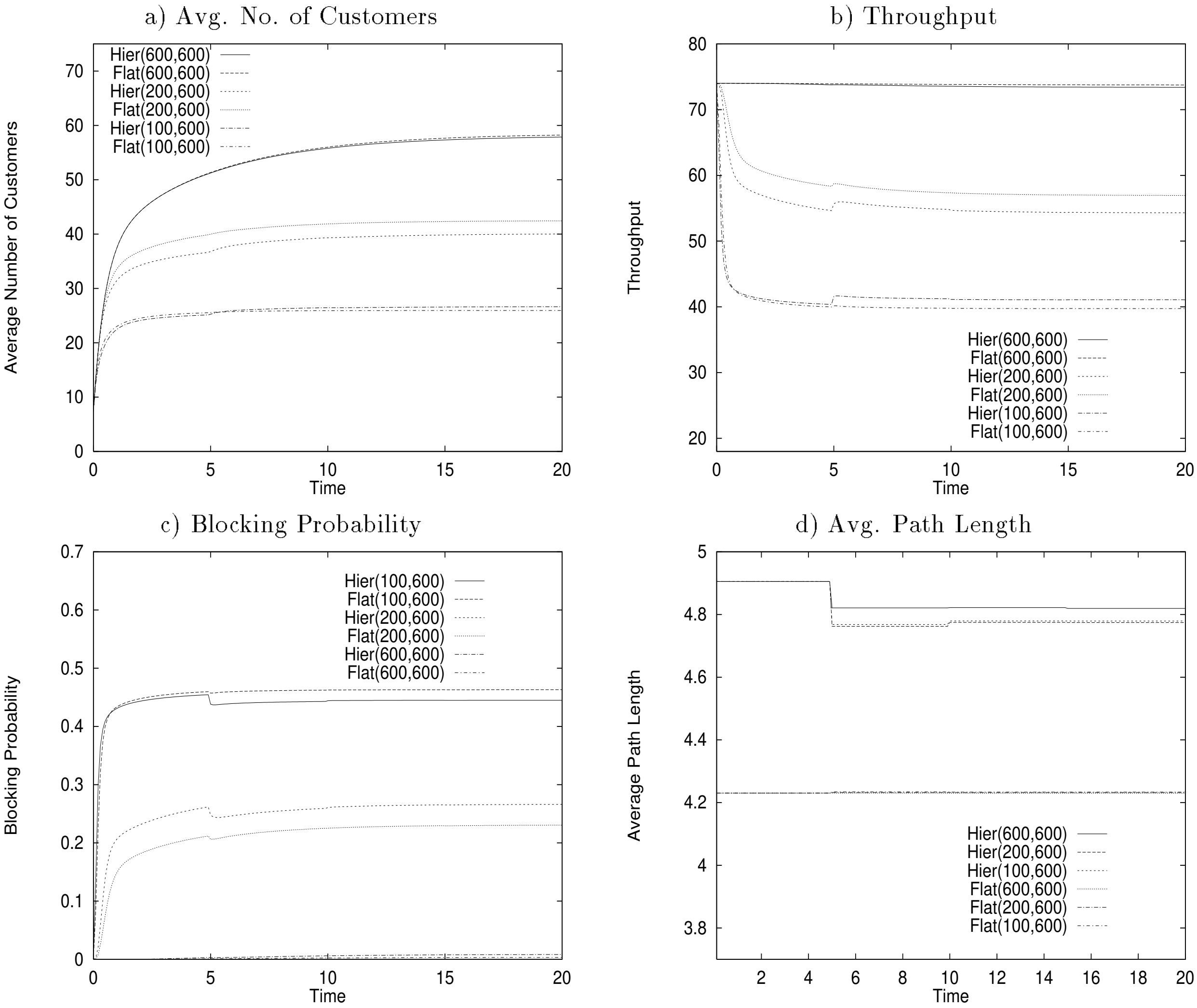

Figure 4: Performance Metrics of Topology 1

3. Varying the ratio of the bandwidth of links inside a level-1 group to those between level-1 groups causes changes in the relative performance of the two schemes.

Validation of the Z-iteration results against simulations are shown in Figures 6 and 6. As can be seen, they agree very well. The simulations took about 4 hrs on a Ultra Sparc whereas the Z-iteration took about 4 minutes. Thus our numerical method is effective at accurately evaluating QoS routing in large networks.

Finally, we compare the memory, computation, and communication costs of hierarchical

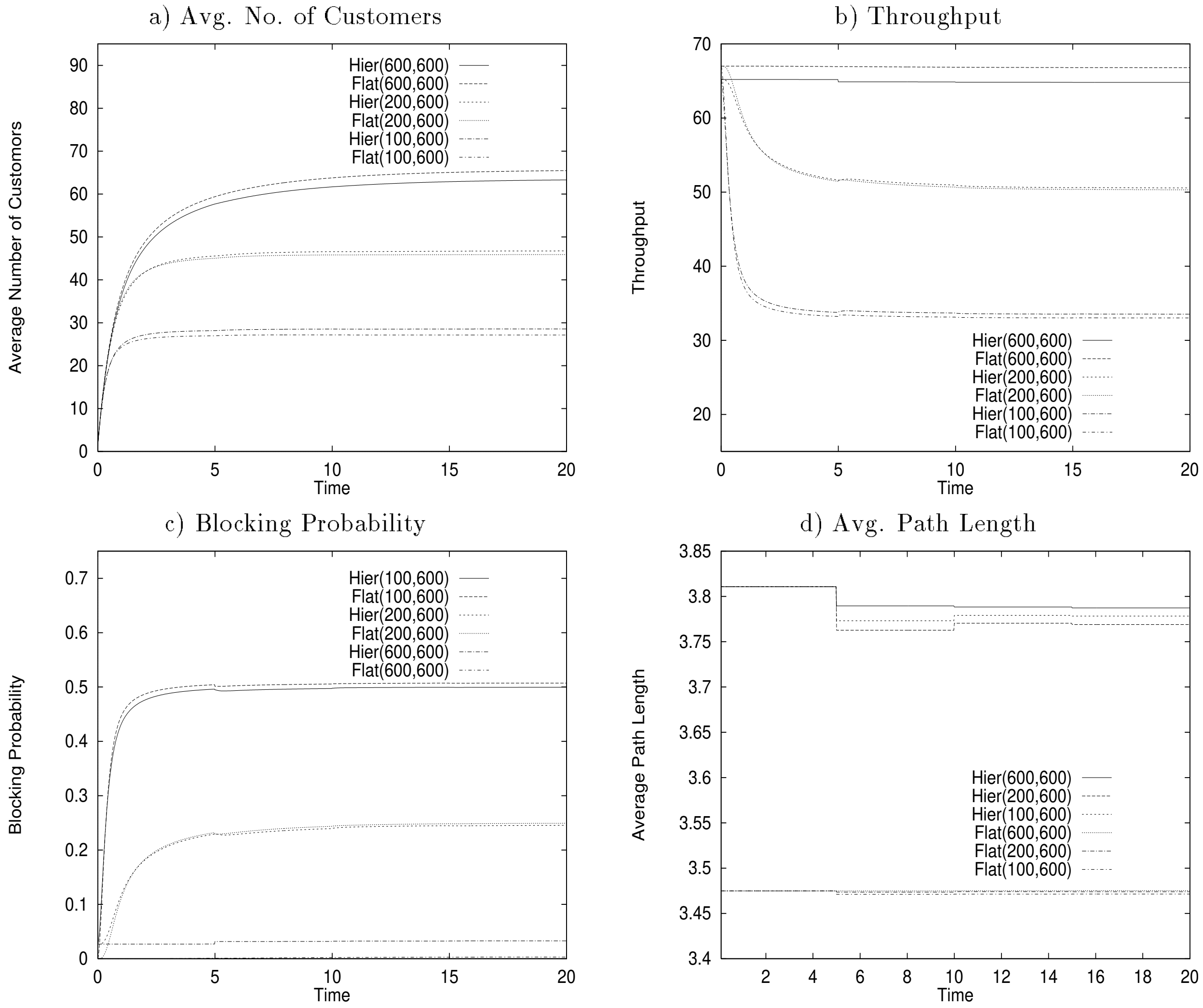

Figure 5: Performance Metrics of Topology 2

and flat routing. The worst case size of routing tables and the number of update messages exchanged at every broadcast are statically determined. Let N be the number of nodes in the network and E be the number of edges. In flat routing each node's view has a maximum size of N^2. In hierarchical routing each node has a view of size at most $((N_1 - 1) + N_0)^2$ where N_1 is the number of level-1 nodes, and N_0 the maximum number of nodes in a group.

To obtain an estimate of the number of messages for every broadcast, we use assume the network has the capacity to broadcast a message to a particular group or to the entire network using an algorithm that sends out the message to every node just once, i.e using

Routing	Hierarchical	Flat	Hierarchical / Flat
Routing Table Size Top 1	4900	256	5.22%
Routing Table Size Top 2	1764	256	14.51%
No of Broadcast Msgs Top 1	4900	819	16.71%
No of Broadcast Msgs Top 2	1764	639	36.22%

Table 3: Routing Table and Broadcast Cost

a minimum spanning tree approach. In the case of flat routing the number of broadcast messages cause by one node is N. So the total number of broadcast message on the network is N^2. In hierarchical routing, the number of messages M would be:

$$M = \sum_{j=1}^{n} (N_0^j)^2 + \sum_j N_0^j + N_1^2$$

where n is the number of groups and N_0^j is the number of routers in group j. The quantity in the first sum represents the number of broadcast messages within a group, each node sends a broadcast message to all the other members of its group. Again we assume that a message reaches router just once. The second sum is the number of broadcast messages generated by each leader as it broadcasts to the rest of the group. This term reduces to N. The third term represents the number of messages generated by the leaders broadcasting to each other. Table 3 shows the results of the calculation. Top 1 and Top 2 refers to Topology 1 and Topology 2.

6 Conclusions

We have developed an efficient numerical approach to evaluate hierarchical QoS routing. Unlike simulation. our approach is scalable to the large networks needed to evaluate hierarchical routing. It also models aggregation and delayed feedback in a straightforward way, unlike analytical methods which are usually too "coarse".

For the two example networks we studied, the hierarchical scheme was comparable in blocking probability to flat routing. In the hierarchical scheme the time for the computation of a route at each node is greatly reduced. This effect is not taken into account in our model. We also ignore the effect of broadcast messages on the network. Accounting for these would yield a noticable improvement in the case of the hierarchical scheme. The traffic in the network will be considerably less than in flat routing, resulting in increased network efficiency. The main reason for any performance difference of the hierarchical scheme in this model is due to the the longer paths computed by the routing protocol.

There are several areas of further study. One area is to examine the the correlation between the performance of hierarchical and the link capacities: how does varying the link capacities affect the performance of hierarchical. We see that in Topology 1 there does seem to be a correlation of some sort. This leads to the question of whether the topology of the network plays any role in this. Other, more aggressive methods of aggregation can be considered; achieving shorter average path lengths would enhance the overall performance of the hierarchical scheme.

References

[1] R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent capacity and its application to bandwidth allocation in high-speed networks. *IEEE J. Select. Areas Commun.*, SAC-9(7):968–981, September 1991.

[2] I. Matta and A.U. Shankar. Fast time-dependent evaluation of integrated services networks. *To appear in Computer Networks and ISDN System – Special Issue on Modeling of Wired and Wireless ATM*, 1998. Preliminary version in Proc. IEEE ICNP '94, 1994.

[3] R. Nagarajan, J. Kurose, and D. Towsley. Local allocation of end-to-end quality-of-service in high-speed networks. In *IFIP TC6 Workshop on Modelling and Performance Evaluation of ATM Technology*, page 2.2, January 1993.

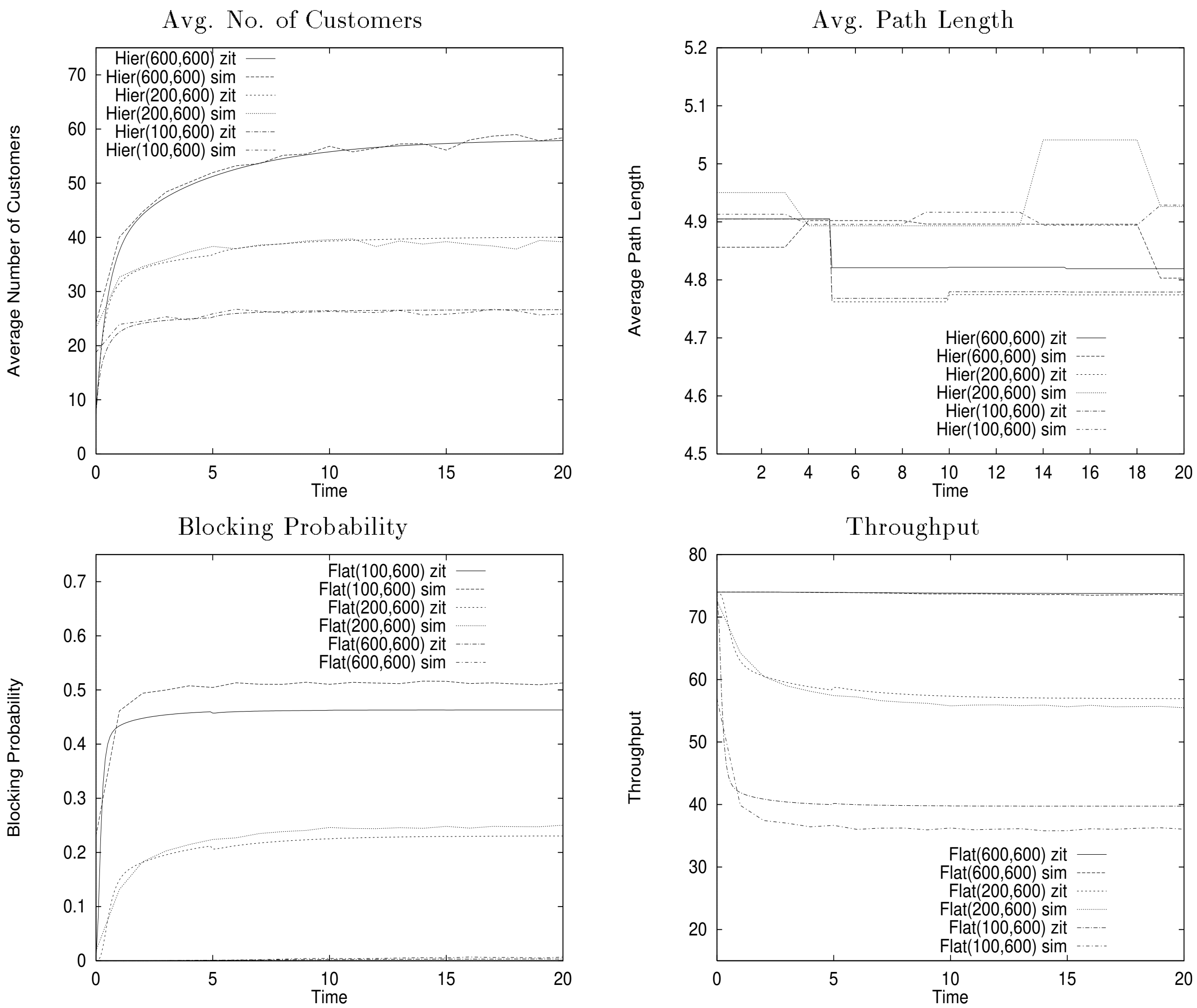

Figure 6: Simulation vs. Z-iteration for Topology 1

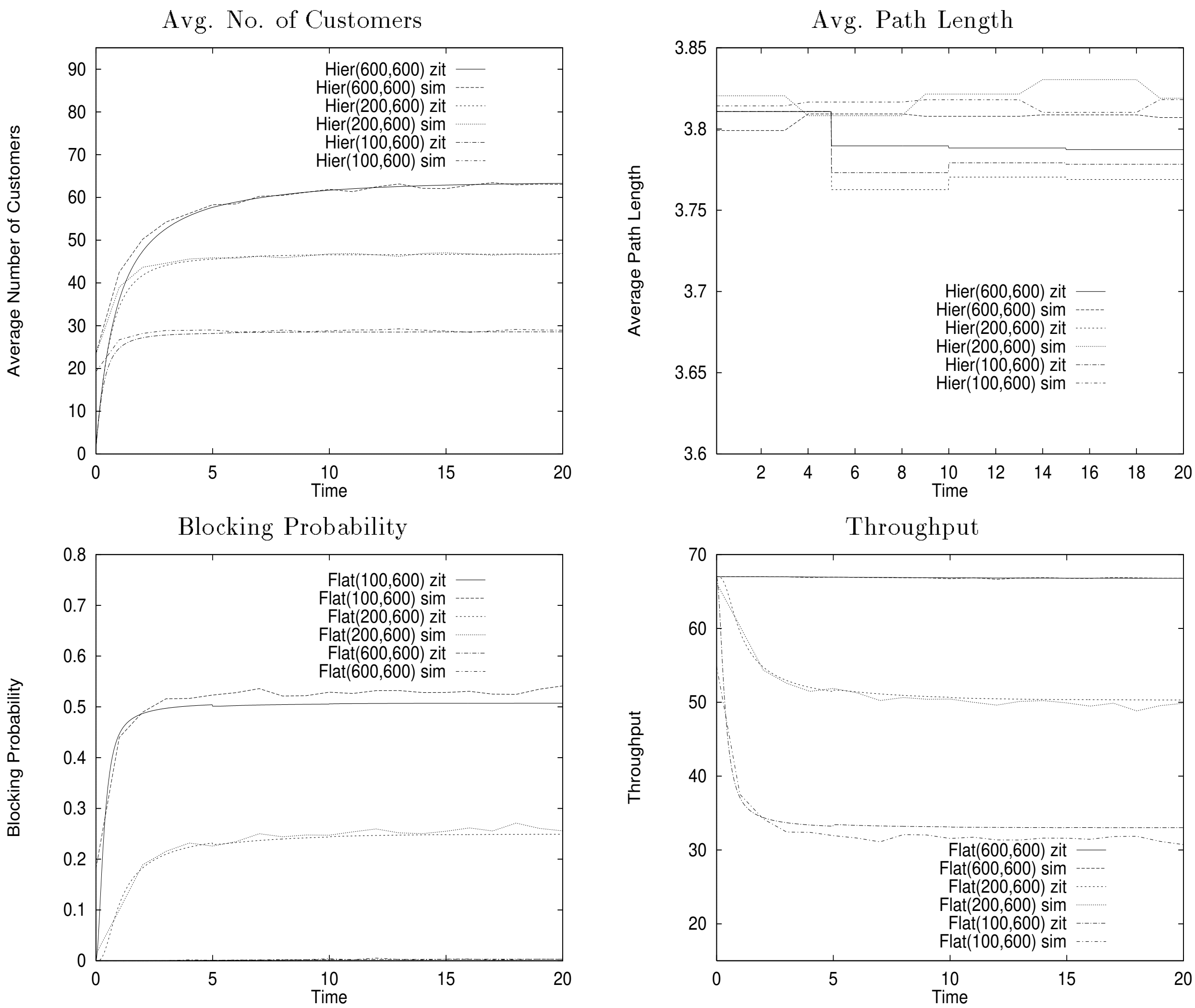

Figure 7: Simulation vs. Z-iteration for Topology 2