A New Path Selection Algorithm for MPLS Networks Based on Available Bandwidth Estimation*

Tricha Anjali[1]**, Caterina Scoglio[1], Jaudelice C. de Oliveira[1], Leonardo C. Chen[1], Ian F. Akyildiz[1], Jeff A. Smith[2], George Uhl[2], and Agatino Sciuto[2]

[1] Broadband and Wireless Networking Laboratory,
School of Electrical and Computer Engineering,
Georgia Institute of Technology, Atlanta, GA 30332 USA
{tricha, caterina, jau, leochen, ian}@ece.gatech.edu
[2] NASA Goddard Space Flight Center,
Greenbelt, MD 20771 USA
{jsmith, uhl, asciuto}@rattler.gsfc.nasa.gov

Abstract. A network should deploy QoS-aware path selection algorithms for efficient routing. To this end, measurements of the various characteristics of the network can provide insight into the state and performance of the network. In this paper, we present a new QoS-aware path selection algorithm for flows requiring bandwidth guarantees based on an estimation algorithm for the available bandwidth on the links of the network. The estimation algorithm for the available bandwidth predicts the available bandwidth and also tells the duration for which the estimate is valid with a high degree of confidence. Thus, the path selection algorithm is adaptive and not very computationally intensive.

Keywords: QoS-aware path selection, Passive measurement, Available bandwidth.

1 Introduction

In recent years, there has been a tremendous growth in the Internet. New applications present new traffic patterns and new Quality of Service (QoS) requirements. A QoS-aware path selection algorithm is needed to improve the satisfaction of the user QoS as well as for improved Traffic Engineering (TE). An efficient path selection procedure enables a network operator to identify, for a new flow, a path with sufficient resources to meet the flow's QoS requirements. These requirements are typically specified in terms of bandwidth guarantees. Thus, the path identification involves some knowledge of the availability of resources throughout the network. Thus, understanding the composition and dynamics of the Internet traffic is of great importance. But observability of Internet traffic is difficult because of the network size, large traffic volumes, and distributed administration. The main tools towards ensuring appropriate QoS satisfaction for all

* This work was supported by NASA Goddard and Swales Aerospace under contract number S11201 (NAS5-01090)

** Corresponding Author. Research Assistant, 250 14th Street, Suite 556, Atlanta, GA 30318, USA Phone: +1-404-894-6616 Fax: +1-404-894-7883

B. Stiller et al. (Eds.): QofIS/ICQT 2002, LNCS 2511, pp. 205–214, 2002.

applications will be measurements of the network and appropriate path selection procedures.

Without loss of generality, we assume in this paper that a source node is presented with a request to reserve bandwidth resources for a new flow with specific QoS requirements, and is responsible for finding a suitable path to the destination. This is similar to the source routing model. Also, we assume the link state information model, where a network topology database is available for state information about nodes and links in the network. The state information is obtained by measurements from the network elements. This information can then be used by the path selection process.

A network can be monitored either in an *active* or *passive* manner. First gives a measure of the performance of the network whereas the latter of the workload on the network. Both have their merits and should be regarded as complementary. The active approach relies on the capability to inject packets into the network and then measure the services obtained from the network. It introduces extra traffic into the network. But the active approach has the advantage of measuring the desired quantities at the desired time. Passive measurements are carried out by observing normal network traffic, without the extra load. The passive approach measures the real traffic. But the amount of data accumulated can be substantial because the network will be polled often for information.

The rest of the paper is organized as follows. In Sect. 2, we present the importance of available bandwidth and a description of various bandwidth measurement techniques. This is followed by a description of our path selection algorithm in Sect. 3. In Sect. 4, we present our algorithm for available bandwidth estimation. In Sect. 5, the results of the experiments with the available bandwidth estimation algorithm are presented. Finally, we conclude in Sect. 6.

2 Measuring Available Bandwidth

There are various quantities of interest that can be insightful about the state of the network. Available bandwidth (together with latency, loss etc.) can predict the performance of the network. Based on the bandwidth available, the network operator can obtain information about the congestion in the network, decide the admission control, perform routing etc. For MPLS networks, the available bandwidth information can be used to decide LSP setup [1], routing (Shortest Widest Path [2], Widest Shortest Path [3]), LSP preemption [4], etc. Several applications could benefit from knowing the available bandwidth of an LSP. One such application is congestion control. Available bandwidth information can also be used to build multicast routing trees more efficiently and dynamically.

It is desirable to obtain the available bandwidth information by measurements from the actual LSPs because they give more realistic information about the available bandwidth. The available bandwidth information can also be obtained by subtracting the nominal reservation for the tunnels from the link capacity which gives a lower bound. In this paper, we will use the terms LSP and link interchangeably because an LSP is equivalent to links in an MPLS network. The path between two LSRs can be composed of multiple LSPs.

The available bandwidth on a link is indicative of the amount of additional load that can be routed over the link. Obtaining an accurate measurement of the available bandwidth can be crucial to effective deployment of QoS services in a network. Available bandwidth can be measured using both active and passive approaches. Various tools to measure available bandwidth of a link in the network are available. The bottleneck bandwidth algorithms can be split into two families: those based on pathchar [5] algorithm and those based on Packet Pair [6] algorithm. The pathchar algorithm is an active approach which leads to the associated disadvantages of consumption of significant amount of network bandwidth etc. The packet pair algorithm measures the bottleneck bandwidth of a route. It can have both active and passive implementations. In [7], the authors have proposed another tool to measure bottleneck link bandwidth based on packet pair technique. None of these tools measures the available bandwidth or utilization of a desired link of a network. In [8], the authors have proposed an active approach to measure the available bandwidth of a route which is the minimum available bandwidth along all links of the path. Another active approach to measure the throughput of a path is Iperf [9] from NLANR that sends streams of TCP/UDP flows. Cisco has introduced the NetFlow technology that provides IP flow information for a network. But in a DiffServ environment, the core of a network is interested in aggregate rather than per-flow statistics, due to the scalability issues.

All the tools, except NetFlow, give path measurements based on an active approach. A network operator, on the other hand, would be interested in finding the available bandwidth on a certain link of the network. He does not need the end-to-end tools that utilize the active approach of measurement. One solution is to use Simple Network Management Protocol (SNMP) which is a short-term protocol to manage nodes in the network. Its operation revolves around two key components: managed devices and management systems. Managed devices store management information in Management Information Bases (MIBs) and make this information available to the management systems. Thus SNMP can be used as a passive technique to monitor a specific device. MRTG [10] is a tool based on SNMP to monitor the network links. It has a highly portable SNMP implementation and can run on most operating systems.

Thus, to obtain the available bandwidth on a certain link of the network in a passive manner whenever he desires, the network manager can use MRTG. But MRTG has the limitation that it gives only 5 minute averages of link utilization. For applications like path selection, this large interval averaging may not be enough. We have modified MRTG to MRTG++, to obtain averages over 10 second durations. This gives the flexibility to obtain very fine measurements of link utilization. We utilize a linear regression-based algorithm to predict the utilization of a link. The algorithm is adaptive because a varying number of past samples can be used in the regression depending on the traffic profile.

3 Path Selection Algorithm

In addition to just finding a path with sufficient resources, other criteria are also important to consider during the path selection. They include optimization of network

utilization, carried load, etc. In the following, we describe the related work followed by the description of the proposed path selection scheme.

3.1 Related Work

Various QoS routing algorithms [3, 11, 12] have been extensively studied in literature. All these schemes utilize the nominal available bandwidth information of the links during the path selection. As most of the traffic flows do not follow the SLA agreement very closely, the nominal utilization of a link is highly overestimated than the actual instantaneous utilization. This leads to non-efficient network resource utilization.

A few measurement based methods are proposed in literature for resource allocation, admission control etc. In [13], a resource allocation method is proposed that estimates the entropy of the system resources based on measurements. A connection admission control algorithm for ATM networks based on the entropy of ATM traffic (obtained from measurements) is given in [14]. Another measurement based admission control framework is given in [15]. A measurement-based routing algorithm is given in [16]. The algorithm tries to minimize the end-to-end delay.

3.2 QoS-aware Path Selection

we propose a path selection scheme that is based on obtaining more accurate link utilization information by actually measuring the same from the network elements involved. However, it is not efficient to update the link state database with the available bandwidth information with very fine granularity, due to scalability concerns. Also, due to the varying nature of the available bandwidth, a single sample of measured available bandwidth does not have much significance. This is because the actual traffic profile may be variable and path selection decisions based on a single sample are more than likely to be wrong. Thus, a method has to be devised that performs an estimation of the link utilization in the future. We have proposed an algorithm for available bandwidth estimation that is explained in next sections.

Once, we obtain the available bandwidth estimates, we can then use these values as the input to a shortest widest path routing algorithm to find an efficient path. Even though there is some inaccuracy involved in path selection based on estimates of available bandwidth, the results in [17] demonstrate that the impact of the inaccuracy in bandwidth information on the path selection procedure is minimal. Our path selection procedure is detailed in Fig. 1. As shown, the available bandwidth estimation algorithm is used only for the links where the estimate has not been calculated before. The parameter $threshold$ in the figure is used as a benchmark for path selection. If the bandwidth request is more than a certain fraction of the bottlenect link bandwidth,the request is rejected. This is done to limit the congestion in the network. Once the estimate is obtained, the available bandwidth information can be used in the shortest widest path routing algorithm as weights of the links. The new flow will be admitted into the network if the flow's bandwidth requirement is less than a certain threshold, which is calculated based on the available bandwidth on the path. Optionally, the arrival of the flow can trigger the re-dimensioning/setup of an LSP between the ingress and the egress LSRs. An optimal policy for this procedure is given in [1].

Path Selection Algorithm:

1. At time instant k, a bandwidth request r arrives between nodes i and j.
2. Run the available bandwidth estimation algorithm links with no bandwidth estimation available.
3. Compute the best path using the shortest widest path algorithm with weights as calculated in step 2.
4. Obtain the available bandwidth A on the bottleneck link of the path.
5. If $r > A * threshold$, reject this path and return to step 3. Else, path is selected for the request.
6. If no path available, request rejected and network is congested.

Fig. 1. The path selection algorithm

4 The Available Bandwidth Estimator for MPLS Networks

Our approach for available bandwidth estimation is based on the use of MRTG. A network operator can enquire each router in the domain through SNMP and obtain the information about the available bandwidth on each of its interfaces. The most accurate approach will be to collect information from all possible sources at the highest possible frequency allowed by the MIB update interval constraints. However, this approach can be very expensive in terms of signaling and data storage. Furthermore, it can be redundant to have so much information. Thus we present an approach that minimizes the redundancies, the memory requirements for data storage and the signaling effort for data retrieval.

We can set the measurement interval of MRTG and measure the average link utilization statistic for that interval. We define for a link between two nodes *i* and *j*:

- C: Capacity of link in bits per sec,
- $A(t)$: Available capacity at time *t* in bits per sec ,
- $L(t)$: Traffic load at time *t* in bits per sec,
- τ: Length of the averaging interval of MRTG,
- $L_\tau[k],\ k \in \mathbf{N}$: Average load in $[(k-1)\tau,\ k\tau]$.

The available capacity can be obtained as $A(t) = C - L(t)$. So, it would be sufficient to measure the load on a link to obtain available bandwidth. Note that we have not explicitly shown the $i - j$ dependence of the the defined variables. This is because the analysis holds for any node pair independent of others. We also define

- p is the number of past measurements in prediction,
- h is the number of future samples reliably predicted,
- $A_h[k]$: the estimate at $k\tau$ valid in $[(k+1)\tau,\ (k+h)\tau]$.

Our problem can be formulated as linear prediction:

$$L_\tau[k+a] = \sum_{n=0}^{p-1} L_\tau[k-n]\, w_a[n] \qquad \text{for } a \in [1,h] \tag{1}$$

Available Bandwidth Estimation Algorithm:

1. At time instant k, available bandwidth measurement is desired.
2. Find the vectors $\boldsymbol{w_a}, a \in [1, h]$ using Wiener-Hopf equations given p and the previous measurements.
3. Find $\left[\hat{L}_\tau[k+1], \ldots, \hat{L}_\tau[k+h]\right]^T$ from (1) and $[L_\tau[k-p+1], \ldots, L_\tau[k]]$.
4. Predict $A_h[k]$ for $[(k+1)\tau, (k+h)\tau]$.
5. At time $(k+h)\tau$, get $[L_\tau[k+1], \ldots, L_\tau[k+h]]^T$.
6. Find the error vector $[e_\tau[k+1], \ldots, e_\tau[k+h]]^T$.
7. Set $k = k+h$.
8. Obtain new values for p and h.
9. Go to step 1.

Fig. 2. The Available Bandwidth Estimation Algorithm

where on the right side are the past samples and the prediction coefficients $w_a[n]$ and on the left side, the predicted values. The problem can be solved in an optimal manner using Wiener-Hopf equations. We dynamically change the values of p and h based on the traffic dynamics.

The available bandwidth estimation algorithm is given in Fig. 2. We define p_0 and h_0 as the initial values for p and h. In step 2 of the algorithm, we need to solve the Wiener-Hopf equations. They are given in a matrix form as $\boldsymbol{R_L w_a = r_a}$, $a = 1, \ldots, h$, *i.e.,*

$$\begin{bmatrix} r_L(0) & \cdots & r_L(p-1) \\ \vdots & \ddots & \vdots \\ r_L(p-1) & \cdots & r_L(0) \end{bmatrix} \begin{bmatrix} w_a(0) \\ \vdots \\ w_a(p-1) \end{bmatrix} = \begin{bmatrix} r_L(a) \\ \vdots \\ r_L(a+p-1) \end{bmatrix}$$

In order to derive the autocorrelation of the sequence from the available measurements, we estimate it as

$$r_L(n) = \frac{1}{N} \sum_{i=0}^{N} L_\tau[k-i] L_\tau[k-i-n]$$

where N affects the accuracy of the estimation, *i.e.*, more samples we consider, more precise the estimation is. The number of samples needed for a given n and N is $(n + N)$. Since the assumption about stationarity of the measurement sequence may not be accurate, we update the values of the autocorrelation every time we change the value of p in step 8 of the algorithm. The solution of the Wiener-Hopf equations will provide w_a that can be used for predicting $\widehat{L}_\tau(k+a)$, $a = 1, \ldots, h$. Since the matrix $\boldsymbol{R_L}$ is symmetric Toeplitz, Levinson recursion can be used to solve for $\boldsymbol{w_a}$ efficiently. For a size p matrix, it involves $(2p^2 - 2)$ multiplications and $(2p^2 - 3p + 1)$ additions.

From the w_a's, we predict $\left[\widehat{L}_\tau[k+1], \ldots, \widehat{L}_\tau[k+h]\right]^T$ using (1). Next step is to obtain an estimate of the available bandwidth for the interval $[(k+1)t, (k+h)t]$. The available bandwidth $A_h[k]$ is given as $A_h[k] = C - max\left\{\widehat{L}_\tau[k+1], \ldots, \widehat{L}_\tau[k+h]\right\}$ for a conservative estimate. As the estimate is valid for a duration longer than the individual sampling times, the path selection scheme is feasible for applications requiring

Interval Variation Algorithm:

1. If $\sigma/\mu > Th_1$, decrease h till h_{min} and increase p till p_{max} multiplicatively.
2. If $Th_1 > \sigma/\mu > Th_2$, decrease h till h_{min} and increase p till p_{max} additively.
3. If $\sigma/\mu < Th_2$, then:
   (a) If $\mu > Th_3 * M_E^2$, decrease h till h_{min} and increase p till p_{max} additively.
   (b) If $Th_3 * M_E^2 > \mu > Th_4 * M_E^2$, keep h and p constant.
   (c) If $\mu < Th_4 * M_E^2$, increase h and decrease p till p_{min} additively.

Fig. 3. Algorithm for h and p

estimates of available bandwidth valid over long durations. Thus, even though LSPs are long-living, our path selection procedure is still valid.

After obtaining the actual load $[L_\tau[k+1], \ldots, L_\tau[k+h]]^T$ at time $(k+h)t$, we find the prediction error vector $[e_\tau[k+1], \ldots, e_\tau[k+h]]^T$ where each element is given as

$$e_\tau[k+a] = \left(L_\tau[k+a] - \widehat{L}_\tau[k+a]\right)^2 \quad \text{for } a = 1, \ldots, h.$$

Next, we estimate new values for p and h based on a metric derived from the mean (μ) and standard deviation (σ) of error e_τ. The algorithm is given in Fig. 3.

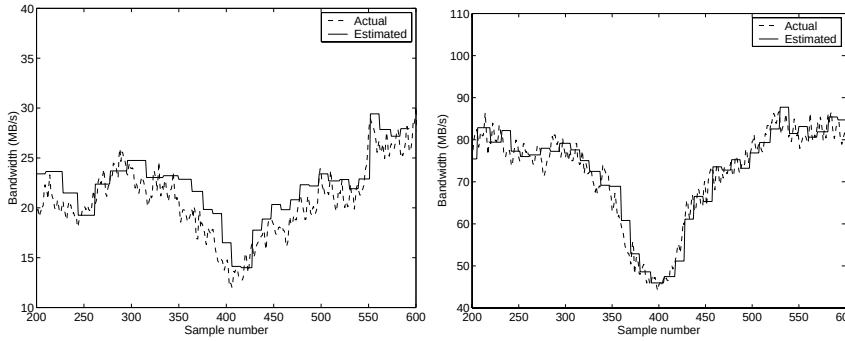
In the algorithm, M_E is the maximum error value and we have introduced h_{min} and p_{max} because small values of h imply frequent re-computation of the regression coefficients and large values of p increase the computational cost of the regression. Also, we have introduced the thresholds Th_1 to Th_4 to decide when to change the values of the parameters p and h. They are determined based on the traffic characteristics and the conservatism requirements of the network domain. The values of these parameters are held constant throughout the network. In other words, they are not decided on a link-by-link basis.

In this section, we have presented an algorithm for estimating the available bandwidth of a link by dynamically changing the number of past samples for prediction and the number of future samples predicted with high confidence. The objective of the algorithm is to minimize the computational effort while providing a reliable estimate of available bandwidth of a link. It provides a balance of the processing load and accuracy. The algorithm is based on the dynamics of the traffic, *i.e.*, it adapts itself.

5 Experimental Results

In this section, we describe the experiments we used to validate and quantify the utility of the available bandwidth estimation algorithm, and thus effectively demonstrating the performance of the proposed path selection algorithm. First, we describe a few implementation details about the estimator.

A manager can retrieve specific information from the device using the Object Identifiers (OIDs) stored in the devices. The in-bound and out-bound traffic counters on the router interfaces can be periodically fetched to calculate the traffic rate. We have modified the MRTG software to reduce its minimum sampling down to 10 seconds. This



(a) Input traffic on Atlanta router (b) Input traffic on Cleveland router

Fig. 4. Estimation performance

provides flexibility to the network manager to obtain more precise and instantaneous measurement results. The network manager should decide the optimal measurement period based on the traffic characteristics, the required granularity for the measured values and the appropriate time-scale of the application utilizing the measured values.

Next, we describe the methodology in running the experiments. The algorithm for available bandwidth estimation on a link does not make any assumption about the traffic models. The available bandwidth estimation algorithm works based on the measurements obtained from the network link. Thus, we do not need a network simulator. Instead, we can apply the algorithm to traffic traces obtained from real networks. In the following, we present the traffic profile predicted by available bandwidth estimation algorithm together with the actual profile. We do not present the predicted available bandwidth profile because that can be calculated by taking the difference of the link capacity and the utilization and thus it does not present significant information, when compared to the predicted utilization.

The choice of the thresholds Th_1, Th_2, etc. and h_{min}, p_{max} used for updating the values of p and h in Section 4 has to be made by the network operator depending on the conservativeness requirements of the network operation. We have obtained the following results by choosing $Th_1 = 0.9$, $Th_2 = 0.7$, $Th_3 = 0.5$, $Th_4 = 0.3$ and $h_{min} = 10$, $p_{max} = 50$. All the traffic traces used in the following results have been obtained from Abilene, the advanced backbone network of the Internet2 community, on March 13, 2002. In Fig. 4(a), the available bandwidth estimation algorithm is applied to the input traffic on the Atlanta router of the Atlanta-Washington D.C. link. In Fig. 4(b), same is done for the input traffic on the Cleveland router from the Cleveland-NYC link. As we can see, in both cases, the utilization estimation obtained by taking the peak prediction provides a conservative estimate. Also, when h_{min} is increased, the estimation becomes worse (see Fig. 5), in the sense that it does not follow the sequence closely but is still very conservative. We propose the use of overestimation as a metric to quantitatively measure the performance of the scheme for available bandwidth estimation. For

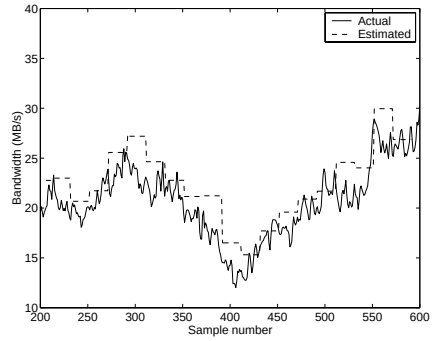


Fig. 5. Input traffic on Atlanta router (h_{min} = 20)

the case in Fig. 4(a), the mean overestimation is 1.31 MB/s for the estimation. Similar values are obtained for the case depicted in Fig. 4(b).

When compared with MRTG, we provide the available bandwidth estimates less frequently without a large compromise in the reliability of the estimate. In other words, the utilization profile obtained as a result of MRTG coincides with the actual traffic profile in Figs. 4(a) and (b), but our algorithm provides an estimate of the link utilization which is nearly accurate. Overall, our path selection algorithm has a higher computational complexity than a simple scheme based on nominal link utilizations. However, in that case, the links will be underutilized as the user agreements are based on the maximum traffic profile.

The performance of the proposed path selection algorithm is better than the shortest widest path routing algorithm because the available bandwidth estimation algorithm provides more accurate estimates of the link available bandwidth than just taking an instantaneous sample. In the worst case, when the link traffic is highly variable, the prediction interval for the available bandwidth estimation algorithm is small. In effect, it is similar to taking instantaneous samples. Thus, the worst case performance of the path selection algorithm is similar to the shortest widest path routing algorithm.

6 Conclusions

We have presented an algorithm for path selection in MPLS networks. The algorithm is based on estimation of the available bandwidths on links of MPLS networks. The available bandwidth estimation algorithm predicts the available bandwidth and tells the duration for which the estimate is valid with a high degree of confidence. The available bandwidth estimation algorithm dynamically changes the number of past samples that are used for prediction and also the the duration for which the prediction holds. The path selection algorithm utilizes these estimates as weights in the shortest widest path routing algorithm. In general, the performance of the path selection algorithm is better than the shortest widest path routing algorithm. In worst case, the path selection algorithm gives a performance similar to shortest widest path routing algorithm.

References

[1] T. Anjali, C. Scoglio, J. de Oliveira, I. Akyildiz, and G. Uhl, "Optimal Policy for LSP Setup in MPLS Networks," *Computer Networks*, vol. 39, no. 2, pp. 165–183, June 2002.

[2] Z. Wang and J. Crowcroft, "Quality-of-Service Routing for Supporting Multimedia Applications," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 7, pp. 1288–1234, September 1996.

[3] R. Guerin, A. Orda, and D. Williams, "QoS Routing Mechanisms and OSPF Extensions," in *Proceedings of 2nd Global Internet Miniconference (joint with GLOBECOM'97)*, Phoenix, USA, November 1997.

[4] J. de Oliveira, C. Scoglio, I. Akyildiz, and G. Uhl, "A New Preemption Policy for DiffServ-Aware Traffic Engineering to Minimize Rerouting,," in *Proceedings of IEEE INFOCOM'02*, New York, USA, June 2002.

[5] V. Jacobson, *Pathchar*, ftp://ftp.ee.lbl.gov/pathchar/, 1997.

[6] S. Keshav, "A Control-Theoretic Approach to Flow Control," in *Proceedings of ACM SIGCOM'91*, Zurich, Switzerland, 1991.

[7] K. Lai and M. Baker, "Nettimer: A Tool for Measuring Bottleneck Link Bandwidth," in *Proceedings of 3rd USENIX Symposium on Internet Technologies and Systems*, San Francisco, USA, March 2001.

[8] M. Jain and C. Dovrolis, "Pathload: A Measurement Tool for End-to-End Available Bandwidth," in *A workshop on Passive and Active Measurements*, Fort Collins, USA, March 2002.

[9] Iperf Tool, *http://dast.nlanr.net/Projects/Iperf/*.

[10] MRTG Website, *http://people.ee.ethz.ch/ oetiker/webtools/mrtg/*.

[11] I. Matta and A. Bestavros, "A Load Profiling Approach to Routing Guaranteed Bandwidth Flows," in *Proceedings of IEEE INFOCOM'98*, San Francisco, USA, April 1998.

[12] K. Kar, M. Kodialam, and T. V. Lakshman, "Minimum Interference Routing of Bandwidth Guaranteed Tunnels with MPLS Traffic Engineering Application," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 12, pp. 2566–2579, December 2000.

[13] P. Barham, S. Crosby, T. Granger, N. Stratford, M. Huggard, and F. Toomey, "Measurement Based Resource Allocation for Multimedia Applications," in *Proceedings of ACM/SPIE MMCN'98*, San Jose, USA, January 1998.

[14] J. T. Lewis, R. Russell, F. Toomey, B. McGuirk, S. Crosby, and I. Leslie, "Practical connection Admission Control for ATM Networks based on On-line Measurements," *Computer Communication*, vol. 21, no. 1, pp. 1585–1596, 1998.

[15] M. Grossglauser and D. Tse, "A Framework for Robust Measurement-Based Admission Control," *IEEE/ACM Transactions on Networking*, vol. 7, no. 3, pp. 293–309, 1999.

[16] N.S.V. Rao, S. Radhkrishnan, and B. Y. Cheol, "NetLets: Measurement-Based Routing for End-to-End Performance Over the Internet," in *Proceedings of International Conferenceon Networking'2001*, 2001.

[17] R. Guerin, "QoS Routing in Networks with Inaccurate Information: Theory and Algorithms," *IEEE/ACM Transactions on Networking*, vol. 7, no. 3, June 1999.