Delay-Constrained Bottleneck Location Estimator and Its Application to Scalable Multicasting

Sangbum Kim and Chan-Hyun Youn

Designing a reliable multicast-based network that scales to the size of a multicast group member is difficult because of the diversity of user demands. The loss inferences of internal nodes by end-to-end measurements do not require the use of complete statistics because of the use of maximum likelihood estimation. These schemes are very efficient and the inferred value converges fast to its true value. In the theoretical analysis, internal delay estimation is possible but the analysis is very complex due to the continuity property of the delay. In this paper, we propose the use of a bottleneck location estimator. This can overcome the analytical difficulty of the delay estimation using the power spectrum of the packet interarrival time as the performance metric. Both theoretical analysis and simulation results show that the proposed scheme can be used for bottleneck location inference of internal links in scalable multicasting.

Manuscript received February 9, 2000; revised November 13, 2000.
This work was partially funded by the Ministry of Information and Communications, Korea, as part of the project 2000-S-001.
The authors are with the School of Engineering, Information and Communication University, Taejon, Korea.
Sangbum Kim (phone: +82 42 866 6126, e-mail: ksb@icu.ac.kr)
Chan-Hyun Youn (phone: +82 42 866 6126, e-mail: chyoun@icu.ac.kr)

I. INTRODUCTION

As the Internet grows, accurate measurements of network dynamics are becoming increasingly important [1]. Mechanisms for accurately measuring network metrics are fundamental to the successful design, control, and management of networks. Distributed real-time applications, such as audio- and video-conferencing, collaborative environments, and distributed interactive simulation require a source which send messages to a selected set of destinations with varying Quality of Service (QoS) delivery constraints [2]. The underlying network must have multicasting and QoS capabilities to efficiently support these applications. There have been many studies which satisfy QoS requirements in multicasting. Unconstrained algorithms build multicast trees in a best-effort way without explicitly accounting for the requested QoS. These algorithms may not provide feasible trees that satisfy the QoS requirements. Thus, constrained algorithms that construct feasible trees with the constrained parameters are more suitable for QoS requirements. Therefore, it is required that the underlying multicast protocol find a QoS–constrained minimum cost tree. However, finding such a tree is known to be computationally expensive [3]. Particularlly, in terms of multicast scalability, the congestion region should be narrower since it is unknown where the multicast members are to join. This means that bottleneck nodes need to be estimated. One way to do this is to use the inference of network-internal loss using Maximum Likelihood Estimation (MLE) [1] and the other is the inference of multicast tree and bottleneck bandwidth [4], [5]. The merit of MLE is that the inferred values of internal nodes quickly converge to actual values within a small error. An advantage of the tree inference scheme is that this algorithm appears to converge to the correct tree with high probability. However, they do infer by using the

multicast probe packets and the number of probe packets is important to converge to the actual value and structure. The estimation of bottleneck location range is limited to loose boundaries.

In this paper, we propose the estimation of bottleneck (heavily congested) node among every internal node using the power spectral analysis. This inference estimator of the bottleneck in multicast tree is based on the spectral difference between input power at an ingress node and output power at an egress node [6], [7]. The proposed estimation scheme needs no probe packet. This property gives application-traffic-independence. The results show that this scheme can be used for bottleneck location inference of internal links. This bottleneck location information gives efficient background for the design of scalable multicasting networks.

II. NETWORK MODEL FOR MULTICASTING

Consider a network using a weighted digraph $N = (V, L)$, where V is the set of nodes and L denotes the set of links corresponding to the set of communication links connecting the nodes. The existence of a link $l = (u, v)$ from node u to v implies the existence of a link $l' = (v, u)$ for any $u, v \in V$, *i.e.*, full duplex in networking terms.

In general, packets originating at some source node $s \in V$ in the network have to be delivered to a set $G \subseteq V - \{s\}$ of destination nodes. We will call G the destination set of multicast group. Multicast packets are routed from s to the destinations in G via the links of a multicast tree $T = (V_T, L_T)$ rooted at s. The multicast tree is a subgraph of N(*i.e.*, $V_T \subseteq V, L_T \subseteq L$) spanning s and the nodes in G. In addition, V_T may contain relay nodes, that is, nodes intermediate to the path from the source to a destination. Let $P_T(s, v)$ denote the path from source s to destination $v \in G$ in the tree T. Then, multicast packets from s to v experience a total delay of $\sum_{l \in P_T(s,v)} D(l)$. The total cost of a tree $T = (V_T, L_T)$ is the sum of the cost of all links in that tree (*i.e.*, $Cost(T) = \sum_{l \in T} C(l)$). This optimum delay and delay variation bounded multicast tree problem is called delay; delay variation; and bounded multicast tree problem. It is formally described as follows [8]–[10]:

Constraints: Given a network $N = (V, L)$, a source node $s \in V$, a multicast group $G \subseteq V - \{s\}$, a link-delay function $D : L \to R^+$, a delay tolerance Δ, and a delay variation tolerance δ, then construct a constrained tree $T = (V_T, L_T)$ spanning $G \bigcup \{s\}$, such that

$$\sum_{l \in P_T(s,v)} D(l) \le \Delta \quad \forall v \in G, \tag{C1}$$

$$\left| \sum_{l \in P_T(s,v)} D(l) - \sum_{l \in P_T(s,u)} D(l) \right| \le \delta \quad \forall v, u \in G. \tag{C2}$$

Constraint (C1) is defined as the source-destination delay constraint, while constraint (C2) is called the inter-destination delay variation constraint. Constraints (C1) and (C2) represent two conflicting objectives, respectively. The delay constraint (C1) dictates that short paths should be used. However, choosing the shortest paths may lead to a violation of the delay variation constraint among nodes that are close to the source and nodes that are far away from it. Consequently, it may be necessary to select longer paths for some nodes in order to satisfy constraint (C2). The problem of finding a feasible tree is one of selecting paths in a way that strikes a balance between these two objectives.

III. QOS CONSTRAINTS OF A MULTICAST APPLICATION

Consider the case in which the multicast members wish to join the tree. There are two ways of determining what node to join in. One is the destination-controlled method (leaf node) and the other is the source-controlled method (root node). Regardless of the joining method, the factor that has the impact on the efficiency of resource utilization and QoS is the existence of a node that cannot meet the bandwidth of QoS requirement. We should first find the nodes that are so severely congested that it is impossible to transmit the multicast packets. Consider the trade-off between the calculation complexity and the efficient utilization of resources. As the scale of a multicast network become larger, the probability that the efficiency of resource utilization and the degree of QoS satisfaction lies in the range that we require becomes smaller.

In [11], a robust algorithm called Packet Bunch Mode (PBM) is proposed, this estimates the bottleneck bandwidth along a unicast path. Probe packets transmitted by the sender include a sequence number, and a time-stamp indicating the transmission time. The packet's arrival time is noted at the receiving end. Recording the difference in arrival times ΔT_r between consecutive packets measures the interarrival time of each packet. The difference in transmission times, ΔT_s, is calculated from the packet time-stamps. PBM produces many bandwidth estimates that coincide with known link speeds, and produces few erroneous results. However, this algorithm has a tendency to misdiagnose a multiple-channel bottleneck link with the presence of considerable delay noise.

The real time video service is the most significant aspect of the multicast service. The quality has to be kept constant during the transmission by both the encoder and the network. Many studies on QoS have been performed which focus on the real-time traffic representation scheme [9]. When the packetized

Moving Picture Expert Group (MPEG) video streams were transmitted over the network passing many node and links, they experienced packet loss and delay. Also, the numbers of packets per frame time are changed under various conditions. Recent studies on the MPEG VBR (Variable Bit Rate) models revealed that the autocorrelation function of the MPEG video traffic is not only exponentially decaying but also periodic [6], [7].

The Markov Modulated Poisson Process (MMPP) or Markov Modulated Bernoulli Process (MMBP) used as a video source model did not capture the characteristics of the video input stream since MMPP or MMBP does not show periodicity in the autocorrelation. Specifically, the periodicity of the MPEG video traffic is due to the choice of the encoding scheme and gives a significant effect on the tail behavior of the queue length distribution for a queueing system. The evaluation method to identify the QoS of video services is not fully established because of the difficulties in estimating traffic characteristics under Internet wide area network[11]–[13].

1. Spectral Characteristics of an MPEG VBR Video in a Multicast Tree

In the Periodic-Markov Modulated Batch Bernoulli Process (P-MMBBP), there is a renewal process and two transition probability matrices to reflect the periodicity. The matrix S governs state transitions of the underlying Markov chain at the renewal epochs and the matrix Q governs state transitions of the underlying Markov chain at the non-renewal epochs. Renewal epoch represents the I-frames input into the packet networks. Non-renewal epoch represents the P-frames input into the packet network. Some results [7] showed that Q matrix governs one step transition probability between the non-renewal epochs and S matrix governs transition between the renewal epochs. The transition by the P-frame generation is only governed by Q matrix. That is, I-frame generation between P-frames generation does not affect the one step transition of the P-frames. The following equations show these results:

$$P_0 Q_{I,1} + P_1 Q_{I,2} = Q_P, \tag{1}$$

$$QSQ = Q^2, \tag{2}$$

Q_P : one step transition probability matrix for P-frame

P_0, P_1 : one step transition probability matrix for I-frame evolution

$Q_{I,1}, Q_{I,2}$: one step transition probability matrix for I-frame evolution to P-frame

Let $X(m)$ be random function, which generates the number of fixed-size packets per frame. Therefore, the autocorrelation function of $X(m)$ is obtained by,

$$R(n) = E[X(m)X(m+n)], \tag{3}$$

where, m and n represent the sample points. Figure 1 represents the exponentially decaying property and the periodicity with period = 12. The GOP structure is IPPPPPPPPPPP. Period 12 means that the above encoding scheme makes one I-frame every 12 frames. The sizes of I-frames are generally larger than P-frames.

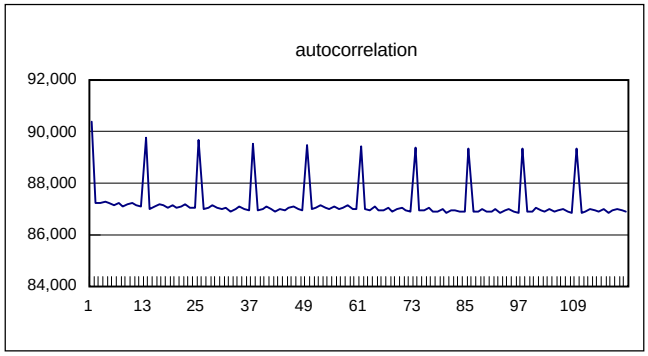


Fig. 1. Autocorrelation of a MPEG video frame size at every frame time with GOP = 12.

Since the packet network can be modeled as a simple system or a cascaded system, we can present a controlled queueing system with input traffic as shown in Fig. 2 [14]. In practice, it is difficult to obtain the exact description of the random input process. Only the statistics are measurable.

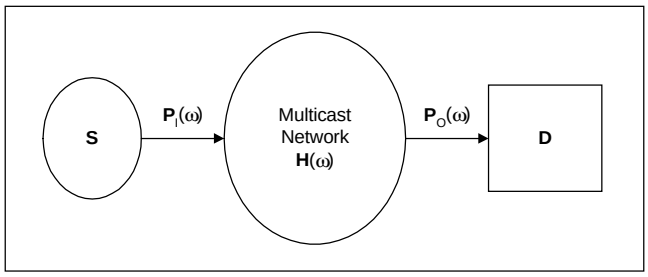


Fig. 2. A multicast network based on a spectral analysis.

The study in [6] and [7] indicates that the queueing performance is much more dependent on second-order input statistics than higher-order ones. As we mentioned previously the second-order statistics are described by power spectral function $P_I(\omega)$ in the frequency domain. Essentially, the low-frequency input power has a dominant impact on queueing performance whereas the high-frequency one can be neglected. This research shows that the traffic in high-speed networks include a large amount of low-frequency bands, which means that the correlation of the traffic is higher. Another point of view is that the

network, especially the buffer, can be viewed as a filter. The buffer in the network eliminates the short-term correlation by storing the fast packet arrival. However, long-term correlation cannot be removed and its property affects the queueing performance. Based on these results, we can observe the output power differences compared to the input power spectrum. This measurement-based scheme can be applied to evaluate the QoS degradation. We emphasize the delay performance varying under diverse network conditions. The packet delay resulting from the network load is simulated and analyzed.

IV. BOTTLENECK LOCATION ESTIMATION

1. Bottleneck Inference Model for a Multicast Network

Combining the information obtained by the bottleneck estimation algorithm and tree structure it is possible to narrow down the possible locations of the bottlenecks in the multicast tree. Receivers appear as leaves in the reconstructed logical tree. The bottleneck bandwidth seen by each interior node is at least equal to the maximum of the bottleneck bandwidth estimates seen by each of its downstream receiver nodes. Contrary to the bottleneck range inference in [2], the inference scheme for the internal node loss (based on multicast network) estimates the probability that a probe packet is transmitted successfully at each node approximated to the actual value.

Let us consider an example of two-leaf logical multicast tree as shown in Fig. 3. In an experiment, a set of probes is dispatched from the source. We consider each probe as a trial, the outcome is a record of whether or not the probe was received. This is expressed in terms of the random process, *X*, each outcome is a set of values of X_k for *k* in the set of leaf nodes *R*, for example, the random quantity $X_{(R)} = (X_k)_{k\in R}$, an element of the space $\Omega = \{0,1\}^R$ of all such outcomes [15].

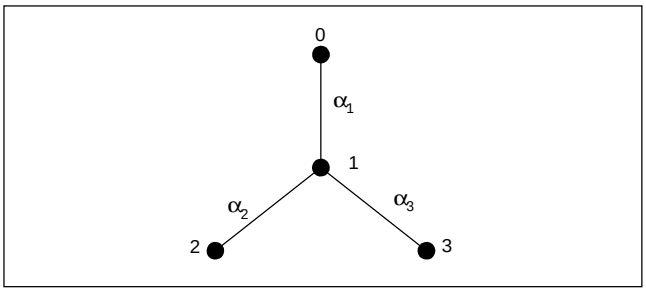


Fig. 3. A two-leaf logical multicast tree.

For a given set of link probabilities $\alpha = (\alpha_k)_{k\in V}$, the distribution of the outcomes $(X_k)_{k\in R}$ will be denoted by P_α. The probability mass function of a single outcome $x \in \Omega$ is $p(x;\alpha) = P_\alpha(X_{(R)} = x)$. If we dispatch *n* probes, then for each possible outcome $x \in \Omega$. Let $n(x)$ denote the number of probes when the outcome *x* is obtained. The probability of *n* independent observations $x^1,\ldots,x^n$ (with each $x^m = (x_k^m)_{k\in R}$) is then

$$p(x^1,\ldots,x^n;\alpha) = \prod_{m=1}^{m} p(x^m;\alpha) = \prod_{x\in\Omega} p(x;\alpha)^{n(x)} \tag{4}$$

Our task is to estimate the value of α from a set of experimental data $(n(x))_{x\in\Omega}$. If we focus on the class of maximum likelihood estimators (MLEs), we estimate α by the value $\check{\alpha}$ which maximizes $p(x^1,\ldots,x^n;\alpha)$ for the data $x^1,\ldots,x^n$.

[Lemma 1] Let $c_i \in (0,1)$, $i = 1,2,\ldots,i_{\max}$, with $\sum_i c_i > 1$. The equation $(1-x) = \prod_i (1-c_i x)$ has a unique solution $x \in (0,1)$.

Proof: Define the notation that $k \le k'$ for $k,k' \in V$ whenever *k* is a descendant of k' or $k = k'$ but $k \ne k'$. It means that a link *k* is at level $l = l(k)$, if there is a chain of *l* ancestors $k < f(k) < f^2(k) < \cdots < f^l(k) = 0$ leading back to the root 0 of *T*. Levels 0 and 1 have only one node. Occasionally, *U* is denoted to $V \setminus \{0\}$. Let $T(k) = (V(k), L(k))$ denote the subtree within *T* rooted at node *k*. $R(k) = R \bigcap V(k)$ will be the set of receivers which are descended from *k*. Let $\Omega(k)$ be the set of outcomes *x* in which at least one receiver in $R(k)$ receives a packet, i.e.,

$$\Omega(k) = \{x \in \Omega : \bigvee_{j\in R(k)} x_j = 1\}. \tag{5}$$

Set $\gamma_k = \gamma_k(\alpha) = P_\alpha[\Omega(k)]$. An estimate of γ_k is

$$\hat{\gamma}_k = \sum_{x\in\Omega(k)} \hat{p}(x), \tag{6}$$

where $\hat{p}(x) := \dfrac{n(x)}{n}$ is the observed proportion of trials with outcome *x*. We will show that α can be calculated from $\gamma = (\gamma_k)_{k\in V}$, and that the MLE

$$\check{\alpha} = \arg\max_{\alpha\in[0,1]^{\#R}} G(\alpha) \tag{7}$$

can be calculated from the estimates $\hat{\gamma}$ in the same manner. The relation between α and γ is as follows. Define $\beta_k = P[\Omega(k) \mid X_{f(l)} = 1]$. The value β_k does obey the recursion

$$\overline{\beta}_k = \overline{\alpha}_k + \alpha_k \prod_{j\in d(k)} \overline{\beta}_j, \quad k \in V \setminus R, \tag{8}$$

$$\beta_k = \alpha_k, \quad k \in R. \tag{9}$$

Then

$$\gamma_k = \beta_k \prod_{i=1}^{l(k)} \alpha_{f^i(k)}. \quad (10)$$

Therefore

$$r_k = A_k, \quad k \in R. \quad (11)$$

Substituting (6) to (8)

$$(1-\gamma_k / A_k) = \prod_{j \in d(k)} (1-\gamma_j / A_k), \quad k \in U \setminus R. \quad (12)$$

Lemma 1 shows that there is a unique case where $A_k > \gamma_k$ which solves (12). We can recover the value α_k uniquely from the value of A_k by taking the appropriate quotients (and setting $A_0 = \alpha_0 = 1$):

$$\alpha_k = A_k / A_{f(k)} \quad k \in U. \quad (13)$$

Candidates for the MLE are solutions of the likelihood equation for the stationary points α of $G(\alpha)$:

$$\frac{\partial G}{\partial \alpha_k}(\alpha) = 0 \quad k \in U. \quad (14)$$

2. Locating Bottlenecks in a Multicast Tree

We identify the multicast tree as comprising the actual network elements (the nodes) and the communication links that join them. The logical multicast tree comprises the branch points of the physical tree and the logical links between them. The logical links consist of one or more physical links. Thus, each node in the logical tree, except for the leaf nodes and possibly the root, must have 2 or more children. We construct the logical tree from the physical tree by deleting all the links with one child (except for the root) and directly joining the parent and child links.

If we consider the logical multicast tree, *T*, we see that this consists of the set of nodes *V*, including the source and receivers, and the set of links *L*, which are ordered pairs (j,k) of nodes that indicate a link from j to *k*. We will denote $U = V \setminus \{0\}$. The set of children of node *j* is denoted by the nodes whose parent is *j*. For each node *j*, other than the root 0, there is a unique node $f(j)$, the parent of j, such that $(f(j), j) \in L$. For each node *j*, we define its level $\ell(j)$ to be the non-negative integer such that $f^{\ell(j)}(j) = 0$. The root $0 \in V$ represents the source of the probes and the set of leaf nodes $R \subset V$ represents the receivers. The basic assumptions are as follows:

- The source traffic has already passed at least one node. The burstiness of the source traffic has been filtered considerably.
- The statistical characteristics at every node are homogeneous.
- The power difference at each node is independent.

The following is the definition of the necessary parameters to infer the bottleneck location. X_k is the power attenuation that occurrs at node k. The power attenuation measured at each interval is a random process and each value measured is independent from the others. We set the threshold to determine whether the node is under the normal conditions for a bottleneck or under a severely congested condition as found in prior results. The determination for the threshold is described in section VI.1. The probability that the random process X_k is less than the threshold, $\Theta(Thd)$ is defined as $\alpha_k(0)$. The complementary probability of $\alpha_k(0)$ is defined as $\alpha_k(1)$.

$$\alpha_k(0): \text{ the probability that } \{X_k < \Theta(Thd)\}, \quad (15a)$$

$$\alpha_k(1): \text{ the probability that } \{X_k > \Theta(Thd)\}. \quad (15b)$$

The end-to-end power attenuation has a cumulative property. Let's consider the power attenuation that occurred at a node. Each node is modeled as a random process, $D_k(n)$, which describes the delay of the n^{th} arrival packet. The function of an arrival time when the n^{th} packet arrives, $T(n)$, is the input function when the traffic is taken into the node. Consider the case that is illustrated in the two-leaf example. Let's call the arrival time measured at the leaf nodes $T_k(n)$. Then $T_k(n)$ is calculated by summing the delay at their passing nodes. Namely,

$$T_1(n) = T(n) + D_1(n), \quad (16a)$$

$$T_2(n) = T(n) + D_1(n) + D_2(n), \quad (16b)$$

$$T_3(n) = T(n) + D_1(n) + D_3(n). \quad (16c)$$

The autocorrelation of the packet interarrival time is derived as follows:

$$R[T_1(n+m)T_1(n)] - R[T(n+m)T(n)] = R[D_1(n+m)D_1(n)], \quad (17a)$$

$$\begin{aligned} &R[T_2(n+m)T_2(n)] - R[T(n+m)T(n)] \\ &= R[D_1(n+m)D_1(n)] + R[D_2(n+m)D_2(n)], \end{aligned} \quad (17b)$$

$$\begin{aligned} &R[T_3(n+m)T_3(n)] - R[T(n+m)T(n)] \\ &= R[D_1(n+m)D_1(n)] + R[D_3(n+m)D_3(n)]. \end{aligned} \quad (17c)$$

The power differences of input and output traffic are the fast Fourier transformations of (17a), (17b), and (17c), respectively. Therefore, the power attenuations at each node are independent

of each other and have a cumulative property. However, we cannot solve (17a), (17b), and (17c) linearly to get the power difference which occurs at each node by end-to-end measurement owing to the number of equations. To overcome this limitation, we need to adopt an estimation theory based on the correlation property between leaf nodes by end-to-end measurement. The cumulative power attenuation from the root node to node k, C_k, can be defined. We also need the level of the tree at node k, ℓ, which is the depth of hierarchy from the root node to node k. C_k is a random process and hence the probability that the value of C_k is located within the threshold is defined as follows:

$$C_k(0): \text{ the probability that } \{C_k < \ell_{\Theta(thd)}\}, \tag{18a}$$

$$C_k(1): \text{ the probability that } \{C_k > \ell_{\Theta(thd)}\}. \tag{18b}$$

[**Theorem**] If $\alpha_k(0)$ is the probability that random process X_k which is the power attenuation that occurs at node k is less than $\Theta(Thd)$ and C_k is the cumulative power attenuation from the root node to node k, $\alpha_k(0)$ is derived from the C_k as follows:

$$\alpha_k(0) = \frac{C_k(0)}{C_{f(k)}(0)}. \tag{19}$$

Proof : Let us define Ω_k as the event that the whole power attenuation is no greater than the threshold of the tree level for at least one leaf node in $R(k)$. The reason to define this is that the correlation of end-to-end bottleneck behavior measured at each leaf node. In other words, these correlations give us clues to estimate the actual values. $\gamma_k = P[\Omega_k]$. γ_k is the result which is measured at each leaf node. $\beta_k(i)$ is the conditional probability that given the cumulative power attenuation up to the parent node of node k, there is at least one node among the leaf nodes of node k where the cumulative power attenuation is under the threshold. $\beta_k(i)$ is needed to calculate the $\alpha_k(0)$ relating the γ_k. $\beta_k(i)$ is defined as below :

$$\beta_k(i): \text{ the probability that } \{\min_{j\in R(k)} C_k(i) \,|\, C_{f(k)}(j)\} \quad i, j = 0, 1 \tag{20}$$

The relation of $\beta_k(0)$ and $\alpha_k(0)$ is

$$\beta_k(0) = \alpha_k(0)\left[1 - \prod_{d\in d(k)}(1-\beta_d(0))\right] \quad k \in U \setminus R \tag{21a}$$

$$\beta_k(0) = \alpha_k(0) \quad k \in R \tag{21b}$$

$$\beta_k(1) = 1 - \beta_k(0) \quad k \in U \tag{21c}$$

Then, by observing that

$$\gamma_k(0) = P\{C_{f(k)} < (\ell-1)_{\Theta(thd)}\} P\{\min_{j\in R(k)} C_k < \ell_{\Theta(thd)} \,|\, C_{f(k)} < (\ell-1)_{\Theta(thd)}\} = C_{f(k)}(0)\ \beta_k(0). \tag{22}$$

We readily obtain

$$\gamma_k(0) = C_k(0)\left[1 - \prod_{d\in d(k)}(1-\beta_d(0))\right] \quad k \in U \setminus R, \tag{23a}$$

$$\gamma_k(0) = C_k(0) \qquad k \in R. \tag{23b}$$

Then, we can compute $C_k(0)$ to obtain $\alpha_k(0)$ as follows:

$$\left(1 - \frac{\gamma_k(0)}{C_k(0)}\right) = \prod_{d\in d(k)}\left(1 - \frac{\gamma_d(0)}{C_k(0)}\right) \quad k \in U \setminus R, \tag{24}$$

$$\gamma_k(0) = C_k(0) \qquad k \in R. \tag{25}$$

Finally,

$$\alpha_k(0) = \frac{C_k(0)}{C_{f(k)}(0)}. \tag{26}$$

Using the theorem, we estimate the bottleneck status of each node based on the spectral attenuation which occurs at each node. The simulation resulting in a two-leaf multicast and a four-leaf multicast network are illustrated in section VI.2.A and VI.2.B, respectively.

A. Example: Four-Leaf Tree

In this section, we discuss the characteristics of the four-leaf tree depicted in Fig. 4. We assume that on each link, a packet suffers power attenuation which is either under the threshold, or over the threshold. Refer to the notations in section V.1.

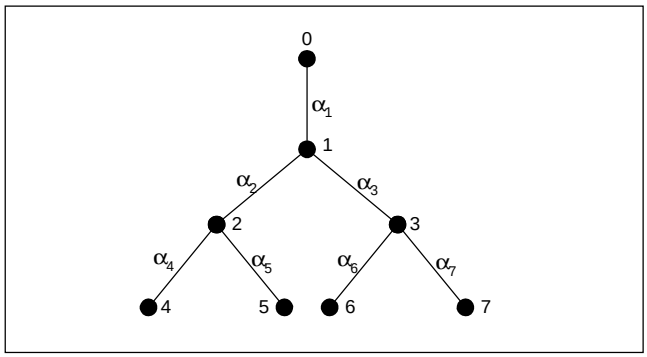


Fig. 4. Four-leaf multicast tree.

In this example, (23a) and (23b) can be solved explicitly by

combining them with (26) to obtain the estimates. The following are the estimates of normal condition probability for each node, from node 1 to node 7.

$$\alpha_1 = \frac{\hat{\gamma}_2(0)\hat{\gamma}_3(0)}{\hat{\gamma}_a}, \tag{27a}$$

$$\alpha_2 = \frac{\hat{\gamma}_4(0)\hat{\gamma}_5(0)}{\hat{\gamma}_b}\frac{\hat{\gamma}_a}{\hat{\gamma}_2(0)\hat{\gamma}_3(0)}, \tag{27b}$$

$$\alpha_3 = \frac{\hat{\gamma}_6(0)\hat{\gamma}_7(0)}{\hat{\gamma}_c}\frac{\hat{\gamma}_a}{\hat{\gamma}_2(0)\hat{\gamma}_3(0)}, \tag{27c}$$

$$\alpha_4 = \frac{\hat{\gamma}_b}{\hat{\gamma}_5(0)}, \tag{27d}$$

$$\alpha_5 = \frac{\hat{\gamma}_b}{\hat{\gamma}_4(0)}, \tag{27e}$$

$$\alpha_6 = \frac{\hat{\gamma}_c}{\hat{\gamma}_7(0)}, \tag{27f}$$

$$\alpha_7 = \frac{\hat{\gamma}_c}{\hat{\gamma}_6(0)}. \tag{27g}$$

where,

$$\hat{\gamma}_a = \hat{\gamma}_2(0) + \hat{\gamma}_3(0) - \hat{\gamma}_1(0)$$
$$\hat{\gamma}_b = \hat{\gamma}_4(0) + \hat{\gamma}_5(0) - \hat{\gamma}_2(0)$$
$$\hat{\gamma}_c = \hat{\gamma}_6(0) + \hat{\gamma}_7(0) - \hat{\gamma}_3(0)\ .$$

V. APPLICATION TO A SCALABLE MULTICASTING TREE FORMATION

One of the advantages of using the bottleneck location estimator is that the overhead of measuring all nodes, which form a multicast tree, is reduced. The tree formation is forward which means the multicast group server knows the status of nodes that comprise the multicast tree. It is not necessary to use the probe packet separately.

A possible method to implement is the preparation of a candidate list. The hop count should be considered. The multicast server collects the statistics that is sent by the receiver sets and manages the gathered information to change the topology considering the QoS. The IP service model deliberately hides the multicast distribution tree. In this scheme, the receivers do not need to know the distribution tree explicitly. The table also provides the hop-scoping to control the distance traveled by multicast packets. The implementation example of the tree formation using the BLE proposed is illustrated in Fig. 5. A normal node and a bottleneck node identify the node. We define the normal node as the QoS-satisfied (especially for delay) node and the bottleneck node as the unsatisfied node. Meeting the QoS requirement means that the delay experienced at each node, is less than the threshold preset to each node.

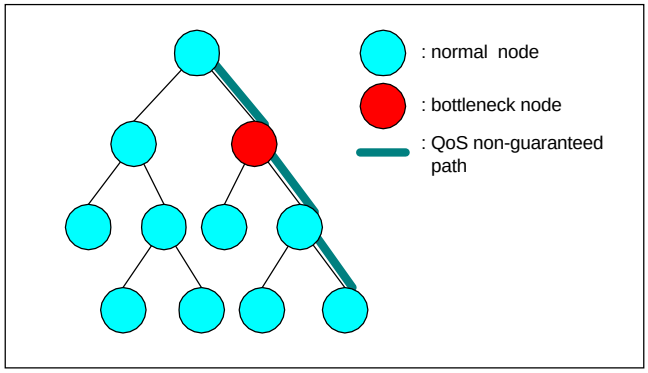


Fig. 5. The example implementation of the tree formation.

```
struct path {
int status[p];        //0 : normal condition, 1 : exceed the threshold
int node[n]};   //stores the path status and status of each nodes on that path
while (1) {
        measure the end-to-end power attenuation periodically;
for (I = 0, i < #_of_paths ; i++) {
  if (end-to-end power attenuation at each leaf node of path i > threshold) {
            path .status[ i ] = 1;
        node = function_of_MLE (measurement);
        }
  else if
        path . status[ i ] = 0;
}
```

Fig. 6. Pseudo code for node identification.

The node identification follows several steps as illustrated in Fig. 6. During the process of tree formation, node identification is required for scalability since the status of every node is needed. Firstly, we need to find out whether the QoS non-guaranteed path exists or not. When the path is in the normal condition, where the cumulative spectral difference is bounded within the threshold we assign, extending the tree following that path can be considered to be reliable. Therefore, we only need to identify the non-guaranteed path. Secondly, the leaf nodes that wish to join the multicast group should not be connected to the same path. Alternative paths will be linked. Next,

we should find the bottleneck node to constrain the tree extensibility. Finally, candidate paths avoiding the bottleneck node are estimated and given to decide what path to join.

VI. SIMULATION RESULTS AND DISCUSSION

1. Basic Models and Assumptions for Simulation

In this section we use the two-leaf tree as the simulation model. There are a few assumptions for the simulation. In section IV, we showed the relation of the power spectral analysis and performance evaluation of the delay property. Based on this result, we will validate our proposed scheme as the foundation for the internal delay estimation. We evaluate these models by the simulator BONeS. The basic assumptions for the simulation are as follows:

- The First In First Out (FIFO) queuing strategy is used.
- All the input streams share the same buffer.
- The size of the buffer is infinite (There is no loss due to buffer overflow).
- There are no losses during the transfer from the source to the destinations. We are only to monitor the delay characteristics.
- Every node has the link capacity of 20 Mbps.
- The generation of the MPEG-video traffic is subject to the P-MMBBP model.
- The MPEG-video traffic has a maximum arrival rate of 9.10 Mbps and the mean arrival rate of 3.64 Mbps.
- The Poisson process governs the generation of the background traffic.

For the simulation, we adopt the upper bound of the 95% confidence interval. According to the sampling of the difference between input and output power spectrum, the upper bound is determined as 2.212. That is,

$$\Theta(Threshold) = 2.212.$$

The 95% confidence interval is also applied for evaluating the cumulative distribution of the power difference at any node. The size of the random sample has an effect on the accuracy of the estimation. In the next section, the results of the probability distribution are illustrated as the sample size increase. The total amount of information in the sample will affect the measure of goodness of the method of inference and must be specified by the experimenter. Specifying a bound on the error of estimation represents this accuracy. The error bound for the 95% confidence interval has a unique value.

2. Simulation Results

A. Two-Leaf Multicast Tree

The two-leaf multicast tree model is shown in Fig. 7.

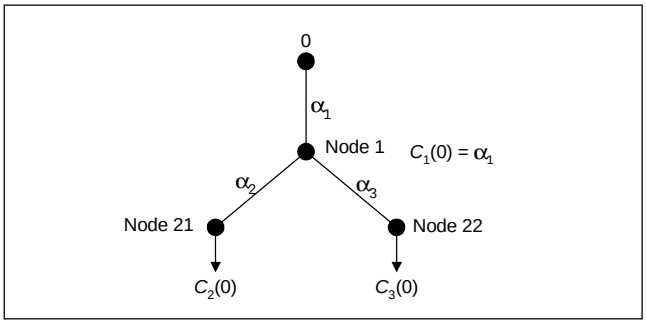


Fig. 7. Two-leaf tree.

In this simulation, we wish to estimate the probability of the power difference at each node, that is, the root node 1 and the leaf node 21 and node 22. We will express the leaf node as our definition. The leftmost number represents the level from the root node 0, then, the following number is the sequence of the node from the left. For example, the leaf node 22 means that it is located in the 2nd level and on the 2nd node in its level. We first calculate the cumulative probability that the end-to-end power difference is under the threshold (determined in section VI.1.). We then evaluate the probability that the whole power attenuation is no greater than the threshold of the tree level for at least one leaf node in $R(k)$. The actual values of α and estimates of α are compared in Table 1. The number of samples is 2000.

Table 1. Comparison of estimates and actual values of probability α.

	Estimates	Actual Values
Node 1 (α_1)	0.940879	0.945286
Node 21 (α_{21})	0.937450	0.935857
Node 22 (α_{22})	0.920162	0.924143

Table 1 shows the information of where the bottleneck occurs. Node 1 gives a node delay to the packets that pass through it. We need another node in order to format a tree that is more reliable regarding the QoS degradation.

B. Four-Leaf Multicast Tree

In this section, we provide the simulation results using a four-leaf tree model. The simulation process is more complex than that of the two-leaf tree. The four-leaf tree consists of seven nodes. The furthest node from the root node 0 is at the level of 3. The iteration starts from the leaf nodes to the root node backward. In this simulation, we need seven ways of measuring. We set the condition of each node as the various environments. The conditions are the background traffic. We assume

that the occupancy of the background traffic at any node is independent of the others. In order to make the difference noticeable, some nodes are set to a very stable condition of offered load while some nodes are set to a seriously-congested situation. In this section, we also examine the analogies and differences between the two-leaf case and the four-leaf case. The first difference to look for when comparing to the two-leaf case is the convergence speed. Generally, more and more random samples are required in order for the model to converge into the region of reliability. The second difference is the error ratio. With the same number of samples, the expanded tree shows the larger error ratio. Using these points of view, we analyze the results of the simulation for the model described in section V.2.A.

The results in the Figs. 8, 9, 10, 11, 12, 13 and 14 show how the estimates converge to the actual values as the sample size increases. The results of node 1, 21, 22, 31, and 32 approximately match the actual values. However, the other nodes reveal the relatively larger error ratio. The following table contains the error ratio with the sample size of 2000.

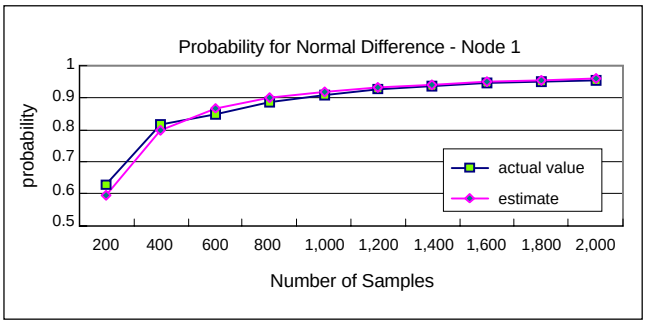


Fig. 8. The actual values and estimates of α_1 vs. the sample size.

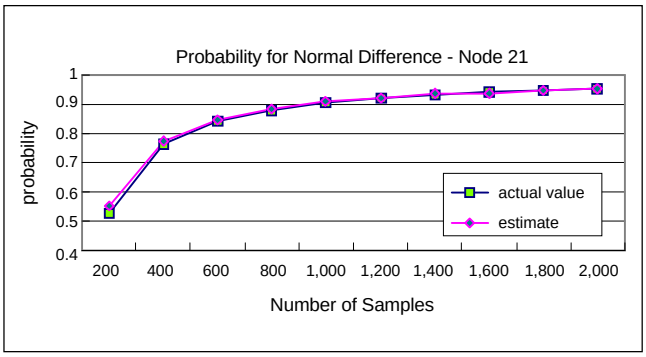


Fig. 9. The actual values and estimates of α_2 vs. the sample size.

Synthesizing the results in Figs. 13 and 14, and Table 2 shows that the larger error relative to the others is due to the correlation property of the nodes. Node 33 is a very unstable node because the probability that the power difference is under normal condition, α_6, is low. Node 34 is very stable since α_7 approximately 1 equals. These results show that the leaf nodes with the same

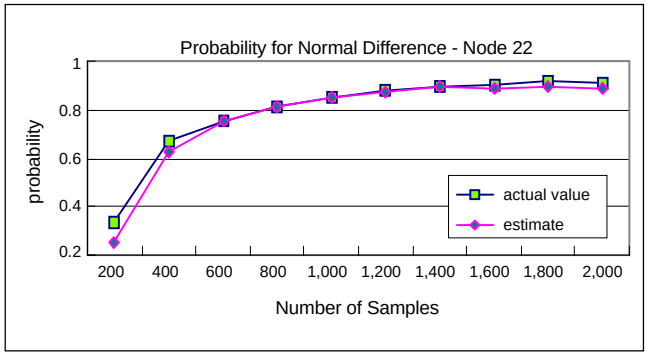


Fig. 10. The actual values and estimates of α_3 vs. the sample size.

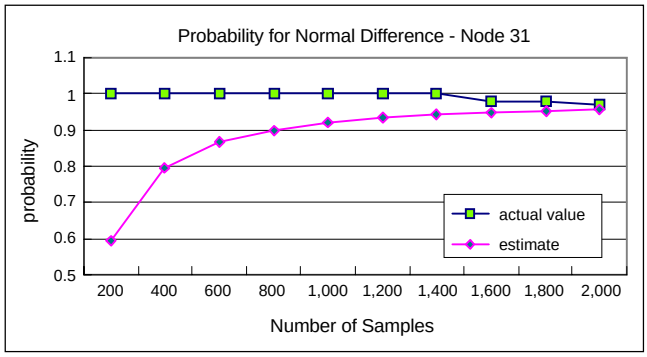


Fig. 11. The actual values and estimates of α_4 vs. the sample size.

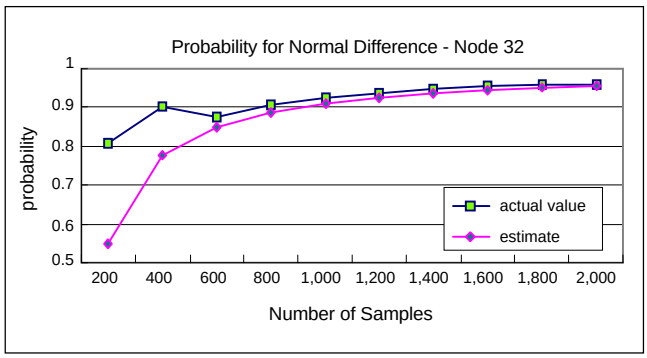


Fig. 12. The actual values and estimates of α_5 vs. the sample size.

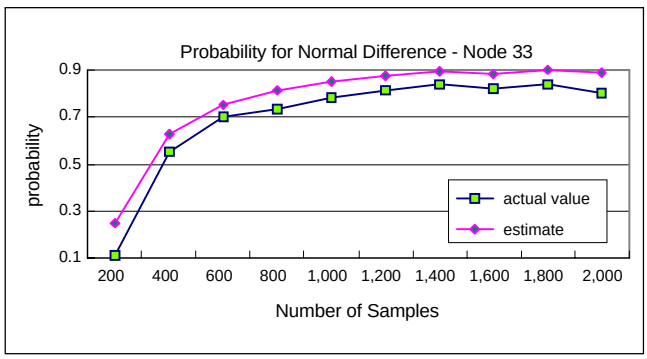


Fig. 13. The actual values and estimates of α_6 vs. the sample size.

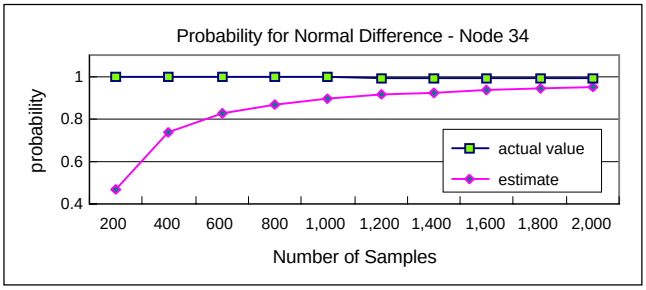


Fig. 14. The actual values and estimates of α_7 vs. the sample size.

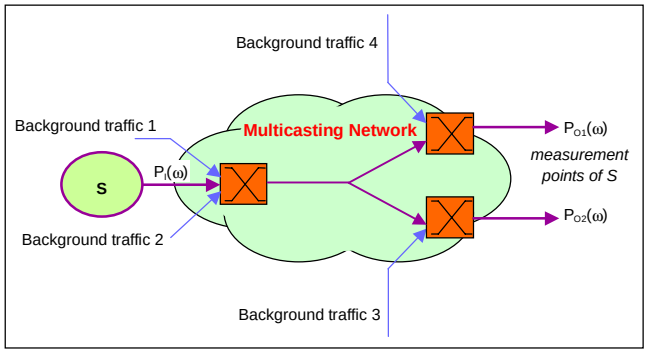


Fig. 15. Conceptual network model of power spectral analysis.

Table 2. Error ratio at each node with sample Size = 2000.

Node	Error Ratio
node 1	0.471204
node 21	-0.05252
node 22	-2.73823
node 31	-1.23839
node 32	-0.625
node 33	10.79226
node 34	-4.43996

parent node may have a larger error if they are under different conditions. This means that more and more random samples are required to make the estimates converge within an allowable error bound.

3. QoS Characteristics of VBR Video at Bottleneck Node

We showed the theoretical analysis of the bottleneck location estimator based on the Maximum Likelihood Estimator. The results are strongly related to the perturbation analysis. The perturbation analysis on power attenuation shows the directional estimate, increasing or decreasing the delay [16]. In this section, we show the QoS evaluation of the VBR video multicasting. Consider the example of the two-leaf tree illustrated in Fig. 15.

The over-perturbed value of the power spectrum variations comes from the power attenuation caused by the delay at the internal nodes. This means that there are a number of packets, which cannot come into the given interval after passing through the network.

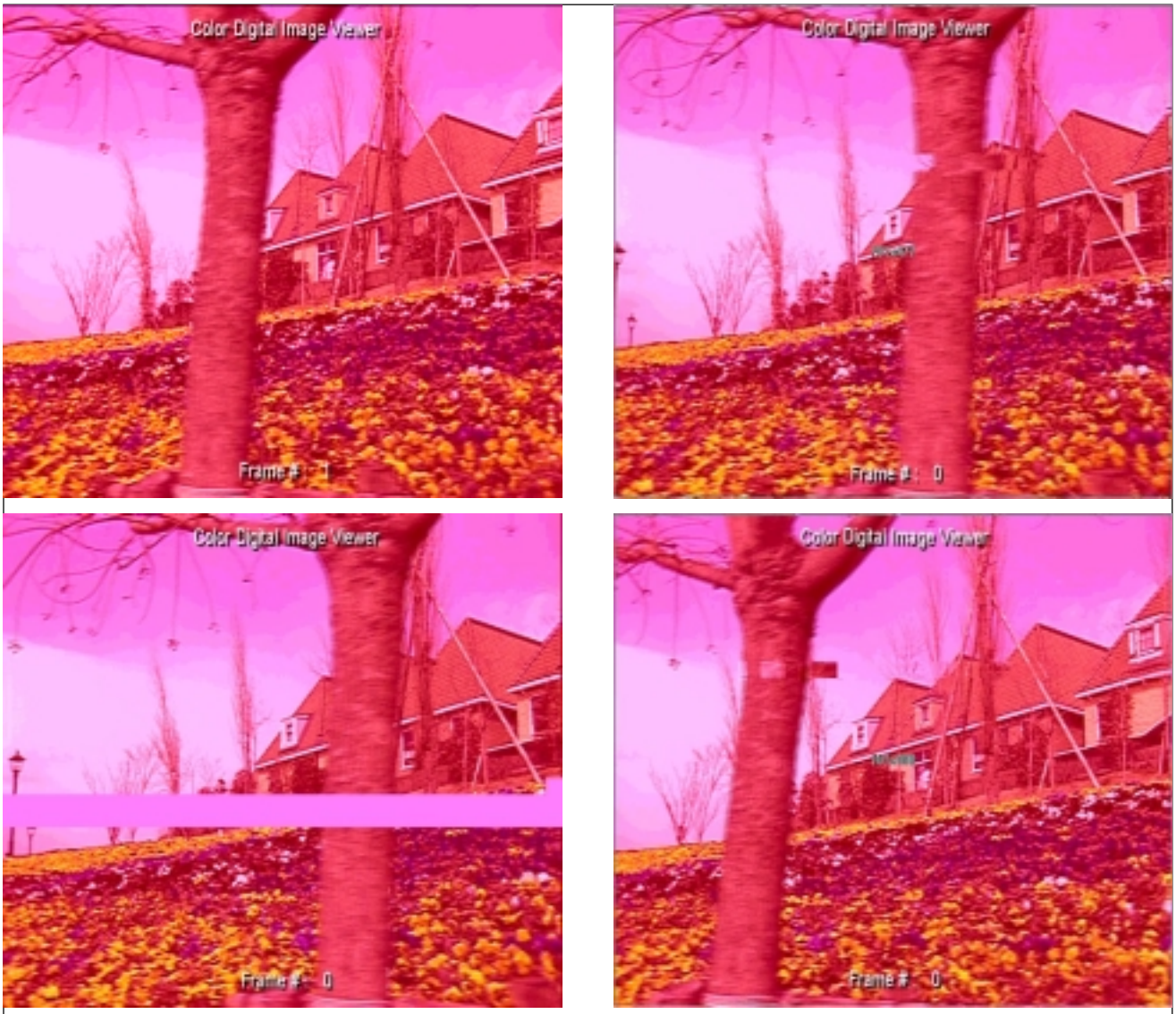
The parameters are as follows for the analytic model. The link capacity is 20 Mbps at each node. The simulation time is 8 seconds. Evaluating the power spectrum is instant and occurs every 1 second. The input video stream has a mean rate of 3.94 Mbps and a maximum rate of 9.20 Mbps. We adopt the P-MMBBP as the video traffic model. All four sources of background traffic flows according to the MMPP source model. The abrupt change of the value implies that the node gets the congested. Note that we could only measure the power spectrum at the ingress and egress node. Only the input power spectrum and the output power spectra at the destination nodes are given and the result at node 1 is unknown. The whole network can be viewed as a set of cascaded filters, which is actually the node. The ingress node on the path to the destination is predominant to transform the spectral characteristics. If we could adjust the traffic load to the adequate range in which the QoS is considered, delay and bottleneck estimation is possible. It is natural that the deviation from the actual value is due to forward nodes. Random function of packet interarrival time is additive and the power spectrum is also additive. If the loads of descendant nodes are significantly different from each other, the power spectra measurements are also different. This implies that the descendant nodes of the ingress node cause a very serious delay. The essential work to be performed is collecting the statistics for looking up in table. The effect of one node measurement is just sufficient to build a table. This data can give the mean delay by a derivative estimation with higher probability. The following results are simulation examples in case that only one of the two-destination node is more delayed than the others.

It is assumed that the ingress node has a stable load. Quality difference between two-destination nodes 2-1 and 2-2 is due to a delay experienced at each node, not due to an ingress node 1. In the case that the load of the two-destination node is low and stable, the quality is the same.

Table 3. Corresponding actual load intensity in Fig. 16.

	Fig.16 (a)			Fig.16 (b)		
	Node 1	Node 2-1	Node 2-2	Node 1	Node 2-1	Node 2-2
Load Intensity	0.64	0.71	0.77	0.74	0.83	0.78

In this section, we proved that this methodology is based on the



(a) Measurement 1 (b) Measurement 2

Fig. 16. Video quality degradation according to bottleneck.

power spectral analysis. The simulation results of video multicast model show that spectral analysis model can follow the delay property. We also have an idea of the dominance of the entrance node into the network. This characteristic is quite useful to infer the actual delay using various estimation schemes and can be applied to scalable multicasting.

VII. CONCLUSION

Designing a reliable multicast that can scale to the size of an audience is a challenge. As the number of receivers grows to thousands or even millions, there is a significant probability of QoS degradation. In this paper, we proposed the bottleneck location estimator based on the performance metric of power spectral analysis. Generally, the power spectral analysis is a function of interarrival time of packets and characterizes input traffic uniquely. Transferring the packet through the network has an influence on the traffic characteristic. The attenuation of the power spectrum means that the QoS of subscribers may be degraded. We can develop an estimator from the spectral perspective. In this paper, we have showed that the Bottleneck Location Estimator based on MLE can find where the congestion occurs by severely measuring the power attenuation. We validate the efficiency of the bottleneck location estimator by simulating the two-leaf and four-leaf multicast tree, respectively. Therefore, as the number of random sample increases, the inferred value of the probability that the power attenuation in advanced node exists under the upper bound. This means that the inferred value con-

verges to the actual value faster and the BLE is applicable to a scalable multicasting scheme.

REFERENCES

[1] R. Caceres, N.G. Duffield, J. Horowitz, D. Towsley, "Multicast-Based Inference of Network-Internal Characteristics: Accuracy of Packet Loss Estimation," *IEEE Infocom* '99, Mar. 1999.

[2] J.H. Lee, "Specification of Communication Services for multimedia cooperative applications based on the reference model of ODP," *ETRI Journal*, Vol. 20, No. 2, June. 1998, pp. 149–170.

[3] T. Ballardie, P. Francis, and J. Crowcroft, "Core based trees: An architecture for scalable inter-domain multicast routing," in *Proc. ACM SIGCOMM*, Ithaca, NY, Sep. 1993

[4] Sylvia Ratnasamy and Steven McCanne, "Inference of Multicast Routing Trees and Bottleneck Bandwidths using End-to-End Measurements," *IEEE Infocom '99*, Mar. 1999.

[5] Vern Paxson, "Towards a Framework for Defining Internet Performance Metrics," *LBNL-38952*, Jun. 1996.

[6] San-qi Li and Chia-Lin Hwang "Queue Response to input correlation functions: Discrete spectral analysis," *IEEE/ACM Trans. Networking*, Vol. 1, No. 5, 1993, pp. 522–533.

[7] Song Chong and San-qi Li "Spectral Analysis of Access Rate Control in High-Speed Networks," *Proc. IEEE Infocom '93*, April 1993, pp. 662–671.

[8] Sung Mo Chung and Chan-Hyun Youn, "Core Selection algorithm for multicast routing under multiple QoS constraints," *IEE Electronics Letters*, Vol. 36, No. 4, Feb. 2000, pp. 378–379.

[9] G.N. Rouskas, and I. Baldine, "Multicast routing with end-to-end delay and delay variation constraints," *IEEE JSAC*, Vol. 15, No. 3, April 1997, pp. 346–356.

[10] S.J. Koh, "Minimizing cost and delay in shared multicast trees," *ETRI Journal*, Vol. 22, No. 1, Mar. 2000, pp. 30–37.

[11] San-qi Li, Sangkyu Park, Dogu Arifler, "SMAQ: A Measurement-Based Tool for Traffic Modeling and Queueing Analysis Part 1: Design Methodologies and Software Architecture," *IEEE Comm. Magazine*, Aug. 1998, pp. 56–65.

[12] P. Pancha, M.E. Zarki, "MPEG coding for variable bit rate video transmission," *IEEE Comm. Mag*., May 1994, pp.54–66.

[13] M. Conti, E. Gregori, and A. Larsson, "Study of the Impact of MPEG-1 Correlations on Video-Sources Statistical Multiplexing," *IEEE JSAC*, Sep. 1996, pp. 1,455–1,471.

[14] San-qi Li, Sangkyu Park, Dogu Arifler, "SMAQ: A Measurement-Based Tool for Traffic Modeling and Queueing Analysis Part 1: Design Methodologies and Software Architecture," *IEEE Comm. Magazine*, Aug. 1998, pp. 56–65.

[15] Andrew Adams, Tian Bu, R. Caceres *et al*, "The use of End-to-End Multicast measurements for Characterizing Internal Network Behavior," *IEEE Comm. Magazine*, May 2000, pp. 152–158.

[16] Chan-Hyun Youn, Sangbum Kim, and Jung-Guk Bae, "Power-Spectral analysis-Based QoS Evaluation of VBR Video and Its Application to Pricing Model," *IEEE IPCCC 2000*, Feb. 2000.

Sangbum Kim received BS in Electronics Engineering from KAIST and MS in Communications Engineering from Information and Communications University (ICU), Taejon, Korea, in 1998 and 2000, respectively. Since 2000, he joined R&D center at Medialink Inc. as an Engineer for the development of switching system hardware. Currently, he is interested in the terabit switch router and optical Internet.

Chan-Hyun Youn received BS and MS degrees in Electronics Engineering from Kyungpook National University, Taegu, Korea, in 1981 and 1985, respectively. He also received a Ph.D. in Electrical and Communications Engineering from Tohoku University, Japan, in 1994. He served at Korean Army as a communications officer, First Lieutenant, from 1981 to 1983. Before joining the University, from 1986 to 1997, he was a leader of high-speed networking team at Korea Telecom (KT) Telecommunications Network Research Laboratories. Where he had been involved in the research and developments of switching system maintenance system, high-speed networking, and HAN/B-ISDN network testbed. Especially, he was a principal investigator of high-speed networking projects including ATM technical trial between KT and KDD, Japan, Asia-Pacific Information Infrastructure (APII) testbed, KOREN and APAN, respectively. Since 1997, he has been an associate professor at Information and Communications University, Taejon, Korea. Currently, he is interested in the high performance routing, multicasting, network performance measurement, and optical Internet. He was a recipient of IEICE PAACS friendship prize, Japan, in 1994.