차세대 네트워크에서의 적응형 절대적 지연 차별화 방식

백 정 훈*

요 약

본 논문에서는 차세대 패킷 네트워크에서의 성능 품질을 개선시키기 위한 하나의 접근 방안으로 패킷 지연에 대한 절대적 차별화 기능을 제공하는 알고리듬을 제안한다. 제안된 알고리듬은 임의 시간 구간에 도착될 입력 트래픽을 예측하고 이를 기반으로 지연 제어 기능을 수행한 후 실제로 입력된 트래픽을 측정하여 예측 오차를 도출하고 이를 다음 시간 구간에서 보상하는 것을 특징으로 한다. 제안된 방식은 매 시간 구간마다 지속적으로 예측 오차를 보상함으로서 예측 편차가 높은 버스트 트래픽에 대하여 특히 우수한 성능을 제공한다. 모의 실험을 통해 제안된 방식은 절대적 성능 지표를 충족하고 기존 방식에 비해 버스트 트래픽에 대하여 우수한 적응성을 제공하는 것을 확인한다.

Adaptive Absolute Delay Differentiation in Next-Generation Networks

Jung Hoon Paik*

ABSTRACT

In this paper, an algorithm that provisions absolute differentiation of packet delays is proposed with an objective for enhancing quality of service (QoS) in future packet networks. It features an adaptive scheme that compensates the deviation for prediction on the traffic to be arrived continuously. It predicts the traffic to be arrived at the beginning of a time slot and measures the actual arrived traffic at the end of the time slot, and derives the deviation between the two quantity. The deviation is utilized to the delay control operation for the next time slot to offset it. As it compensates the prediction error continuously, it shows superior adaptability to the bursty traffic as well as the constant rate traffic. It is demonstrated through simulation that the algorithm meets the quantitative delay bounds and shows superiority to the traffic fluctuation in comparison with the conventional mechanism.

Key words : Quality of Service

---

* 동아방송대학 정보통신과

1. Introduction

Two broad paradigms for quality-of-service(QoS) in the Internet have emerged, namely integrated services(IntServ) and differentiated services(DiffServ) [1, 2]. The IntServ model, which aims to provide hard end-to-end QoS guarantees to each individual data flow, requires per-flow-based resource allocation and service provisioning and, thus, suffers from the scalability and manageability problems due to the huge amount of data flows.

This lack of scalability is, to a large extent, being addressed within the DiffServ architecture. In the DiffServ model, traffic is aggregated into a finite number of service classes that receive different forwarding treatment. It achieves scalability and manageability by providing quality per traffic aggregate and not per application flow. However, it's drawback is difficulty in contriving efficient resource allocation mechanisms to guarantee the end-to-end QoS of each individual data flow.

With superiority in terms of scalability and manageability, the DiffServ is gaining more popularity as the QoS paradigm for the future Internet. Several schemes are devised to realize the DiffServ philosophy. At one end of the spectrum, absolute differentiated services seek to provide end-to-end absolute performance measures without per-flow state in the network core[3]. At the other end of the spectrum, relative differentiated services seek to provide per-class relative services[4]. In this model, the traffic from a higher priority class will receive no worse service than the traffic from a lower priority class.

In our view, absolute differentiated service is essential for handling a real-time application which requires guaranteed QoS measures for future Internet. In addition, proportional differentiated service is also needed to handle the soft-real time service which is tolerant to occasional delay violations and hence do not require strict delay bounds.

Consequently, it is perceived that the QoS architecture that provides any mix of absolute and relative differentiated schemes under the DiffServ paradigm is the most suitable service architectures for future Internet.

In this paper, an algorithm that enforces absolute differentiation of packet delays is proposed. In [5], Joint Buffer Management and Scheduling (JoBS) scheme is suggested, and it provides relative and absolute per-class service differentiation for delays and loss rate. It makes predictions on the delays of backlogged traffic, and uses the predictions to update the service rate of classes and the amount of traffic to be dropped. Our approach is similar to [5] in that it predicts delays of backlogged traffic and uses the predictions to update the service rate of classes, but main difference is whether the prediction error which occurs indispensably is applied on future control operation. While most conventional schemes don't reflect the prediction error, our algorithm makes use of the deviation to improve the QoS quality. More specifically, it predicts traffic to be arrived at the beginning of a time slot and also measures the actual arrived traffic at the end of a time slot. The prediction deviation is derived at the beginning of a next time slot, and it is quantified to be reflected to the delay control mechanism for the next time slot. The target delay is adjusted by some extent which is determined by the prediction error at every time slot. As the suggested algorithm continually compensates the prediction error every time slot, it shows superior adaptability to the bursty traffic as well as the constant rate traffic as compared with conventional approaches.

The remainder of this paper is organized as follows. In Section 2, related work is overviewed. In Section 3, an algorithm which provisions the quantitative differentiated services is developed. Following this, in Section 4, a set of simulation experiments to illustrate the performance of the scheme is presented. Finally, in Section 5, some concluding remarks are presented

2. Related Work

In DiffServ architecture, an admission control scheme is mainly used to provide QoS guarantees by reserving appropriate resource[6]. There are two basic approaches to admission control. The first, which is called parameter-based approach, computes the amount of network resources required to support a set of flows given a priori flow characteristics. The second approach which is measurement-based relies on measurement of actual traffic load in order to make admission decisions. Measurement-based approaches are classified to two schemes, envelopes-based [7, 8], and probing-based[9].

In [10] and [11], the definition of a statistical bound on arriving traffic is employed to obtain the statistical multiplexing gain in a single node with a packet scheduling algorithm under the scalability constraint.

In [12], the probing rate at a receiver is used as the admission condition. The loss probability of probing packets is used as a threshold to admit or reject a flow in [13].

Relative delay differentiation is first discussed in detail in [14]. In [14], two packet schedulers that try to achieve proportional delay differentiation is presented. However, the schedulers are not ideal, in the sense that, the average delays experienced by different classes tend to deviate from the proportional model under light traffic loads.

Joint Buffer Management and Scheduling (JoBS) is suggested in [5], and provides relative and absolute per-class service differentiation for delays and loss rate. It makes predictions on the delays of backlogged traffic, and uses the predictions to update the service rate of classes and the amount of traffic to be dropped.

In [15], extended weight fair queueing (WFQ) is devised and applied to proportional delay differentiation service. It shows that the delay requirements can be achieved efficiently.

A new scheduler, Deadline Fair Sharing (DFS), is suggested in [16]. It operates in a dynamic weighted fair manner to provide an absolute delay guarantee and proportional delay and loss differentiation guarantees.

Probing mechanism which is incorporated into the EEAC-SV scheme is devised to enhance the end-to-end QoS granularity in the DiffServ network [17].

3. Adaptive Delay Differentiation Model

3.1 Objective

It is assumed that there are N service classes, and class I +1 is better than class i for $2 \leqq i \leqq N$, in terms of service metrics. With this convention, the service guarantees for the classes can be expressed. An absolute delay guarantee on class i is specified as

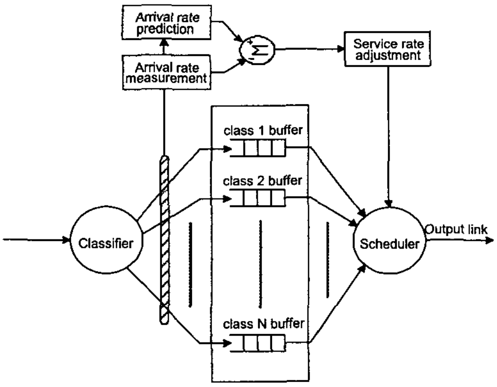
$$D_i = D_i^{target}, \quad \forall i \in \{1, ..., M\} \qquad (1)$$

where D_i^{target} is a desired delay bound of class i. The proportional delay guarantee between class i and class i+1 is defined as

$$\frac{D_{i+1}}{D_i} = \alpha_i^{target}, \quad \forall i \in \{M+1, ..., N\} \qquad (2)$$

where α_i^{target} is a constant that quantifies the proportional differentiation desired.

3.2 Node Architecture

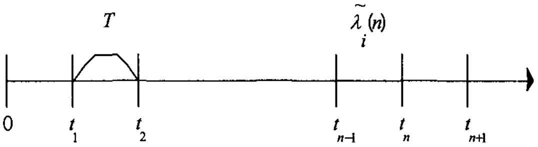


(Figure 1) The proposed system architecture

The proposed node architecture is shown in (Figure 1). The classifier classifies incoming traffic into a number of classes and the scheduler then serves traffic in class buffers. Input traffic is predicted at the beginning of the time slot and measured at the end of the time slot, and the difference will feed into a process to adjust the service rate in the scheduler periodically.

3.3 Service Rate Adjustment

As illustrated in (Figure 2), time axis is slotted with interval T, and time slot n spans the time interval $[t_n-1, t_n]$.



(Figure 2) Time axis notation

The input rate $\tilde{\lambda}_i(n)$ of class i for the time slot n is predicted with the weighted moving average schemes like equation (3) with $\rho = 0.9$. Specifically, predicted values are indicated by a tilde(~).

$$\tilde{\lambda}_i(n) = (1-\rho)\frac{\sum_{k=n-N-1}^{n-2} \lambda_i(k)}{N} + \rho\lambda_i(n-1) \qquad (3)$$

The backlog $B_i(t)$ of class i at time t is derived from $R_i^{in}(t)$ and $R_i^{out}(t)$ like equation (4) where $R_i^{in}(t)$ is the arrived traffic at class i buffer and $R_i^{out}(t)$ is the serviced traffic from class i buffer in the interval [0, t] respectively.

$$B_i(t) = R_i^{out}(t) - R_i^{in}(t) \qquad (4)$$

Now, some parameters related a class i are predicted to derive the service rate for the next time slot n. With the predicted input rate for the next time slot n of equation (3), the prediction of the class i input traffic for next time slot n, $\widetilde{R_i}^{in}(t; t \in [t_{n-1}, t_n])$, is given by

$$\widetilde{R}_i^{in}(t;\ t \in [t_{n-1}, t_n]) = \widetilde{\lambda}_i(n) \times (t - t_{n-1}) \tag{5}$$

Similarly, with the definition of service rate $\gamma_i(n)$ of class i buffer for next time slot n, the predicted serviced traffic of class buffer i for next time slot n, $\widetilde{R}_i^{out}(t;\ t \in [t_{n-1}, t_n])$, is given by

$$\widetilde{R}_i^{out}(t;\ t \in [t_{n-1}, t_n]) = \gamma_i(n) \times (t - t_{n-1}) \tag{6}$$

With the equation (5) and (6), the predicted backlog $\widetilde{B}_i(t; t \in [t_{n-1}, t_n])$ of class buffer i for next time slot n is derived as

$$\begin{aligned} &\widetilde{B}_i(t;\ t \in [t_{n-1}, t_n]) = \\ &B_i(t_{n-1}) + \{\lambda_i(n) - \gamma_i(n)\} \times (t - t_{n-1}) \end{aligned} \tag{7}$$

Now, the predicted delay $\widetilde{D}_i(t;\ t \in [t_{n-1},\ t_n])$ of an class i input packet arriving at time t, $t \in [t_{n-1},\ t_n]$, is described as equation (8).

$$\begin{aligned} &\widetilde{D}_i(t;\ t \in [t_{n-1}, t_n]) \\ &= \frac{\widetilde{B}_i(t;\ t \in [t_{n-1}, t_n])}{\gamma_i(n)} \\ &= \frac{B_i(t_{n-1}) + \{\widetilde{\lambda}_i(n) - \gamma_i(n)\} \times (t - t_{n-1})}{\gamma_i(n)} \end{aligned} \tag{8}$$

Averaging the instantaneous delay $\widetilde{D}_i(t)$ over a time slot n provides a simple measure for the history of delays experienced by typical class i packets. It is given by equation (9).

$$\begin{aligned} &\widetilde{D}_i^{avg}(n) = \frac{1}{T}\int_{t}^{t_n} \widetilde{D}_i(x)dx \\ &= \frac{B_i(t_{n-1}) + \frac{T}{2}\{\widetilde{\lambda}_i(n) - \gamma_i(n)\}}{\gamma_i(n)} \end{aligned} \tag{9}$$

It is a feature of our algorithm that the prediction error on the input rates over time slot n is reflected on the derivation of the service rates over next time slot n+1. In order to reflect the prediction error on the input rates on the derivation of the service rates, the error $\triangle\lambda_i$ between the measured input rates $\lambda_i(n)$ and the predicted input rates $\widetilde{\lambda}_i(n)$ is defined as equation (10).

$$\Delta\lambda_i(n) = \lambda_i(n) - \widetilde{\lambda}_i(n) \tag{10}$$

With the definition of equation (10), the delay difference $\Delta D_{i,\Delta\lambda_i}(n)$ caused by the prediction error $\Delta\lambda_i$ on input rates is derived from equation (9) and given by equation (11).

$$\Delta D_{i,\Delta\lambda_i}(n) = \frac{T}{2} \times \frac{\Delta\lambda_i(n)}{\gamma_i(n)} \tag{11}$$

The actual averaged delays D_i^{avg} over time slot n is adjusted with that extent of equation (11) and expressed as equation (12).

$$\begin{aligned} D_i^{avg} &= \widetilde{D}_i^{avg} + \Delta D_{i,\Delta\lambda_i} \\ &= \frac{B_i(t_{n-1}) + \frac{T}{2}\{\widetilde{\lambda}_i\ (n) + \Delta\lambda_i(n) - \gamma_i(n)\}}{\gamma_i(n)} \end{aligned} \tag{12}$$

With the derivation of equation (12), the delay difference ΔD_i from the target delay D_i^{target} over time slot n is given by

$$\begin{aligned} &\Delta D_i(n) = D_t^{target} \\ &- \frac{B_i(t_{n-1}) + \frac{T}{2}\{\widetilde{\lambda}_i(n) + \Delta\lambda_i(n) - \gamma_i(n)\}}{\gamma_i(n)} \end{aligned} \tag{13}$$

As equation (13) indicates the deviation from the desired delay over the time slot n, it is compensated by that extent over next time slot n+1 such as equation (14).

$$D_i^{target}(n+1) = D_i^{target} + \Delta D_i(n) \tag{14}$$

In case that the actual delay is two larger than the target delay, there is a possibility that $D_i^{target}(n+1)$ might become negative in equation (14). As delay cannot be negative, it is fixed at zero for the case such as equation (15).

$$D_i^{target}(n+1) = \begin{cases} 0 & D_i^{target}(n+1) \leqq 0 \\ 2D_i^{target} - \dfrac{B_i(t_{n-1}) + \frac{T}{2}\widetilde{\lambda_i}\ (n) + \Delta\lambda_i(n) - \gamma_i(n)}{\gamma_i(n)}. & oherwise \end{cases} \tag{15}$$

As the target delay for the time slot n+1 is derived, the service rates for the time slot n+1 is derived from the equation (9). It is given by equation (16).

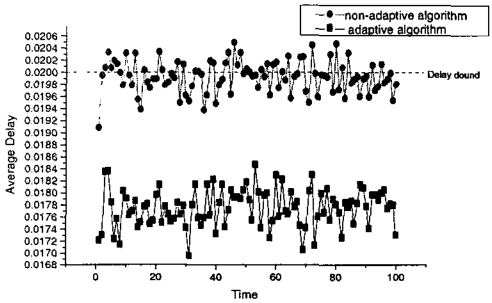
$$r_i(n+1) = \frac{B_i(t_n) + \frac{T}{2}\widetilde{\lambda_i}(n+1)}{D_i^{target}(n+1) + \frac{T}{2}}$$

$$\begin{cases} \dfrac{B_i(t_n) + \frac{T}{2}\widetilde{\lambda_i}(n+1)}{\frac{T}{2}} & D_i^{target}(n+1) \leq 0 \\ \dfrac{B_i(t_n) + \frac{T}{2}\widetilde{\lambda_i}(n+1)}{2D_i^{target} - \frac{1}{r_i(n)}\left\{B_i(t_{n-1}) + \frac{T}{2}[\widetilde{\lambda_i}(n) + \Delta\lambda_i(n) - r_i(n)]\right\} + \frac{T}{2}} & oherwise \end{cases} \tag{16}$$

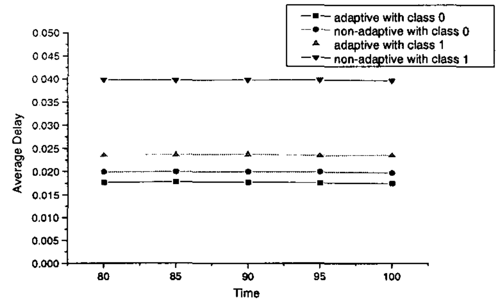
4. Simulation

Simulations for the examination of efficiency and comparisons between adaptive scheme and non-adaptive scheme have been conducted in this section with OPNET simulator. Each source node generates number of traffic flows whose time inter-arrival and packet size are exponentially distributed with mean 0.001 and 1000bits. We create two absolute service classes 0 and 1, and two proportional classes 2 and 3.The higher number of classes has higher priority. The delay requirements are set to 20ms and 40ms respectively for absolute delay, and $\alpha_3 = 0.5$(delay ratio is 1 : 2) for proportional delay. Traffic load distribution is set to 30%, 20%, 30%, and 20% respectively in order to observe the dependency of delay bound on the amount of traffic generated. Link capacity is set to 100 Mbps. We also assume that the link propagation delay is negligible.

(Figure 3) shows a result of queueing delay of class 1. It shows that the adaptive scheme clearly meets the delay bound while non-adaptive scheme frequently exceeds the delay bound.

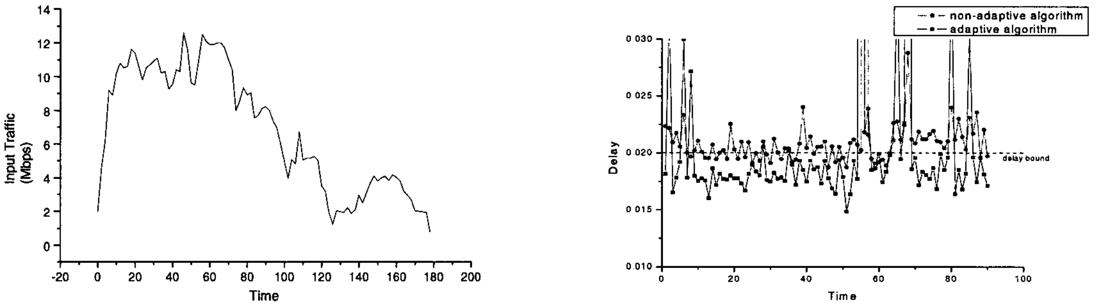


(Figure 3) The queueing delay of the class1



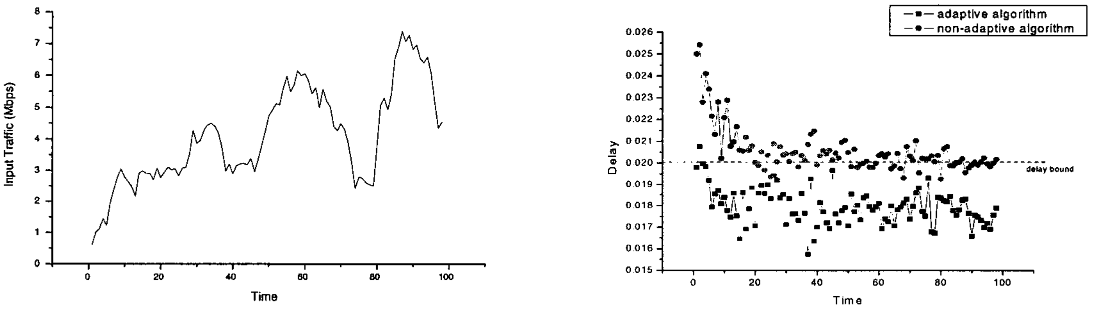
(Figure 4) The effect of link utilization on delay

In addition, we increase the link utilization by giving more number of flows to the proportional elay classes (class 3 and 4) and see the effect of the link utilization. As shown in (Figure 4), the



(a) Bursty input traffic

(b) The delay to bursty traffic I



(c) Bursty input traffic II

(d) The delay to bursty traffic II

(Figure 5) The delay to bursty traffic

link utilization does not affect the absolute delay.

Comparing average queueing delay, adaptive algorithm has lower delay than the target delay while the non-adaptive algorithm is almost same as the target delay. This explains that the non-adaptive algorithm can be more preferable in terms of resource utilization issue.

Though this is true for exponential traffics, we point out that many of current Internet traffics do not follow the exponential distribution. Our adaptive algorithm has better performance for such a realistic traffic. For realistic traffics, we create hundreds of flows which follow Pareto distribution with shaper value 1.9, and starting time is uniformly distributed with specific time period. In addition, the duration of flows follows Pareto distribution with location of 20 seconds and shaper value of 1. 9. The example of the input traffic is shown as (Figure 5). We simulate two algorithms using these traffic and results are shown. In this scenario, the advantage of our algorithm is obvious. Most of average delay does not meet the delay boundary of 20ms in conventional scheme while more number of delays meets the bound in our algorithm. This superior adaptability to burst traffic is originated from the continual compensation of predicted traffic which is a key to our algorithm.

5. Conclusion

In this paper, a delay differentiation algorithm that achieves absolute QoS provisioning is proposed. The main feature of this algorithm is that it continually adjusts the target delay with reference to the traffic prediction deviation in previous time section.

It has founded that the suggested scheme performs well in terms of achieving absolute QoS provisioning. In addition, it shows superior adaptability to the traffic fluctuation in comparison with conventional approach, and it presents a feasible approach to future Internet where QoS differentiation is essentially required and bursty traffic is prevailed.

References

[1] S. Shenker and J. Wroclawski, "General Characterization Parameters for Integrated Service Network Elements", IETF RFC 2215, September 1997.

[2] S. Blake, D. Black, M, Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated service", IETF RFC 2475, December 1998.

[3] I. Stoica and H. Zhang, "Providing Guaranteed Services Without Per Flow Management", ACM SIGCOMM'99, Cambridge, MA, September 1999.

[4] C. Dovrolis, D. Stiliadis, and P. Ramanathan, "Proportional Differentiated Services : Delay Differentiation and Packet Scheduling", ACM SIGCOMM'99, Cambridge, MA, September 1999.

[5] L. Liebeherr and N. Christin, "Buffer Management and Scheduling for Enhanced Differentiated Service", Technical Report: University of Virginia, CS-2000-2.

[6] C. Cetinkaya and E. Knightly, "Egress admission control", Proc. IEEE INFOCOMM 2000, Vol.3, No. 1, pp. 1471-1480, Tel Aviv, Israel, March 2000.

[7] C. Cetinkaya, V. Kanodia, and E. Knightly, "Scalable Services via Egress Admission Control", IEEE TRANSACTION ON MULTIMEDIA Special Issue on Multimedia over IP, Vol. 3, No. 1, March 2001.

[8] J. Qiu and E. Knightly, "Measurement-based admission control with aggregate traffic envelopes", Proc. 10th IEEE ITWDC'98, Vol. 9, No. 2, pp. 199-210, Ischia, Italy, Sept. 1998.

[9] L. Breslau, E. Knightly, S, Schenker, I. Stoica, and H. Zhang, "Endpoint admission control: Architectural issue and performance", ACM SIGCOMM 2000, Vol. 30, No. 4, pp. 57-69, Stockholm, Sweden, Aug. 2000.

[10] R. Boorstyn, A. Burchard, J. Liebeherr, and C. Oottamakorn, "Statistical service envelopes : Statistical bounds on multiplexed traffic in packet network", Proc. IEEE INFOCOMM 2000, Vol. 3, No. 1, pp. 1223-1232, Tel Aviv, Israel, March 2000.

[11] R. Boorstyn, A. Burchard, J. Liebeherr, and C. Oottamakorn, "Statistical service assurances for traffic scheduling algorithm", IEEE. J. Sel. Areas Commun., Vol. 18, No. 12, pp. 2651-2664, Dec. 2000.

[12] G. Bianchi, A. Capone, and C. Petrioli, "Throughput analysis of end-to-end measurement-based admission control in IP", Proc. IEEE INFOCOM 2000, Vol. 3, No. 1, pp. 1461-1470, Tel Aviv, Israel, March 2000.

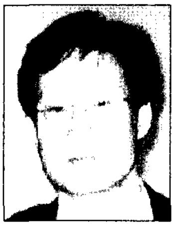
[13] V. Elek, G. Karlsson, and R. Ronngren, "Admission control based on end-to-end measurements", Proc. IEEE INFOCOM 2000, Vol. 2, No. 1, pp. 623-630, Tel Aviv, Israel, March 2000.

[14] C. Dovrolis, D. Stiliadis, and P. Ramanathan, "Proportional Differentiated Services: Delay Differentiation and Packet Scheduling", ACM SIGCOMM'99, Cambridge, MA, September 1999.

[15] Chin-Chang Li, Shiao-Li Tsao, Meng Cheng Chen, Yeali Sun, and Yueh-Min Huang, "Proportional Delay Differentiation Service Based on Weighted Fair Queueing", 2000 IEEE Globecomm, pp. 418-423.

[16] Duc-Long Pham, Shinji Sugawara, and Tetsuya Mike, "A New Scheduler for Real-time Applications in Differentiated Services Networks", IEEE Globecomm, pp. 2841-2846. 2005.

[17] Jie Yang, Jian Ye, Symeon Papavassiliou, Nirwan, and Ansari, "A Flexible and Distributed Architecture for Adaptive End-to-End QoS Provisioning in Next-Generation Networks", IEEE Journal on Selected Areas in Communications, Vol. 23, No. 2, February 2005.



백 정 훈

1986년 한양대학교 전자공학과 (공학사)
1988년 한양대학교 전자공학과 (공학석사)
1999년 한양대학교 전자공학과 (공학박사)
2002년~현재 동아방송대학 정보통신과 교수