Adaptive Multi-level Streaming Service using Fuzzy Similarity in Wireless Mobile Networks

Chong Deuk Lee[1*]

[1]Div. of Electronic Engineering, Chonbuk National University

무선 모바일 네트워크상에서 퍼지 유사도를 이용한 적응형 멀티-레벨 스트리밍 서비스

이종득[1*]
[1]전북대학교 전자 공학부

Abstract Streaming service in the wireless mobile network environment has been a very challenging issue due to the dynamic uncertain nature of the channels. Overhead such as congestion, latency, and jitter lead to the problem of performance degradation of an adaptive multi-streaming service. This paper proposes a AMSS (Adaptive Multi-level Streaming Service) mechanism to reduce the performance degradation due to overhead such as variable network bandwidth, mobility and limited resources of the wireless mobile network. The proposed AMSS optimizes streaming services by: 1) use of fuzzy similarity metric, 2) minimization of packet loss due to buffer overflow and resource waste, and 3) minimization of packet loss due to congestion and delay. The simulation result shows that the proposed method has better performance in congestion control and packet loss ratio than the other existing methods of TCP-based method, UDP-based method and VBM-based method. The proposed method showed improvement of 10% in congestion control ratio and 8% in packet loss ratio compared with VBM-based method which is one of the best method.

요 약 무선 모바일 네트워크 환경에서 스트리밍 서비스는 불확실한 동적 채널 속성으로 인하여 매우 중요한 이슈가 되고 있다. 특히 혼잡, 지연, 지터 등과 같은 오버헤드는 적응형 멀티-레벨 스트리밍 서비스의 성능 저하를 가져오고 있다. 본 논문에서는 무선 모바일 네트워크상에서의 가변 대역폭, 이동성 자원 제약 등으로 인한 성능 저하를 줄이기 위하여 AMSS 메카니즘을 제안한다. 제안된 AMSS는 다음과 같은 목적을 두고서 스트리밍 서비스를 최적화하는데 있다. 1) 퍼지 유사도 척도 이용, 2) 버퍼 오버플로우와 자원 소모로 인한 패킷 손실 최소화, 3) 혼잡과 지연으로 인한 패킷 손실 최소화. 시뮬레이션 결과 제안된 기법이 기존의 TCP-기반 기법, UDP-기반 기법, 그리고 VBM-기반 기법에 비해서 혼잡 제어와 패킷 손실율의 성능이 더 우수함을 보인다. 제안된 기법은 성능이 우수한 VBM-기반 기법과 비교해 볼 때 혼잡 제어율은 10%, 그리고 패킷 손실율은 8%의 성능 향상을 보였다.

Key Words : Wireless mobile network, Congestion, Jitter, Buffer overflow, Packet loss

1. Introduction

Streaming media service is becoming one of the major issues in wireless mobile networks. For the currently deployed mobile nodes, the practical data rates are not enough to support full rate, high quality streaming services. As a result, many research efforts have been proposed to reduce the tradeoff between the high demand of service quality and the limited wireless communication resources among users.

To enhance QoS of the streaming media, the multi-level technique need to be performed to meet the stringent resource constraints at lower level. For very low bit rate channels, providing the priority of stream objects at high quantization distortion levels is appropriate to users. A better solution is to decide the sequence of streaming using fuzzy similarity. Typically, streaming media services in wireless mobile networks are realized mainly by using RTP (Real-Time Transport Protocol) or RTCP (Real-Time Control Protocol). A RTCP packet consists of 5 packets including SR (Sender Report) packet and RR (Receiver Report) packet. SR and RR are used to monitor RTT (Round Trip Time), jitter, and packet loss [1, 2]. This method, however, has the difficulty in adaptive monitoring of packet loss and delay and in periodic monitoring of received packet information. Adaptive streaming methods are proposed to solve these difficulties and to improve service performance and stream quality in wireless mobile applications [3, 4].

The P2P-based streaming service method is relatively efficient for multimedia streaming. In P2P-based mobile applications, a client, as a peer, performs streaming of data packets from other clients as well as to other clients. A client must help the server by the cooperation with other clients. The performance of the system can be improved by alleviating the burden of the server through the cooperation of clients. In distributed mobile applications, however, peers cannot control data packets adaptively due to limited resources.

Thus, this paper proposes a new AMSS for the adaptive streaming of large size multimedia data. The proposed AMSS optimizes streaming services by: i) Use of fuzzy similarity metric, ii) Minimization of packet loss due to buffer overflow and resource waste, and iii) Minimization of packet loss due to congestion and delay. Simulation showed that the proposed method has better performance than the other existing methods of TCP-based streaming method and UDP-based streaming method.

2. Related works

2.1 TCP-based streaming method

Recently TCP (Transmission Control Protocol) is used for multimedia streaming in the wireless mobile network. TCP-based protocols, however, have problems of frequent disconnection and packet loss due to delay and congestion [7]. TCP is not suitable as the transport protocol for streaming media due to its lack of control on the delay and its frequent deep fluctuation of the sending rate, especially when the required end-to-end delay of the multimedia application is small. It is also difficult for TCP-based streaming to work in a multicast environment. In addition, throughput efficiency is of a concern for TCP-based streaming in the wireless mobile network. TCP-based streaming method is efficient when the available bandwidth is sufficiently large and the delay time is short. However TCP does perform congestion control, but this control creates large fluctuations in the fill rate in the receiver buffer. This is far from optimal multimedia traffic, since a typical media traffic flow is highly sensitive to sudden and large rate changes.

2.2 UDP-based streaming method

Congestion control in UDP (User Datagram Protocol)-based multimedia streaming has to take care of not only the fairness and responsiveness of the protocol, but also the rate smoothness to achieve better playback quality of the media applications. A number of TCP-friendly congestion control methods for streaming media have been proposed to provide smoother sending rate [8]. These include the window-based methods and rate-based methods which can be further classified into the probed-based and equation-based methods [9, 10]. The equation-based congestion control methods can achieve good TCP-friendliness by adapting the sending rate according to the throughput equation of the TCP flows under packet size, packet loss rate, and RTT. TFRC (TCP-friendly Rate Control) is proposed for unicast flows with constant packet sizes in [11]. Several variants of TFRC have been proposed to meet multimedia flows with variable packet sizes. However, these do not consider the end-to-end delay constraint and stream constraint of the multimedia applications.

2.3 VBM-based streaming method

A VBM (Virtual Buffer Management)-based streaming

method influences in burst time and media size. So, its bandwidth requirement is highly variable, which it difficult to achieve efficient playback quality. Bandwidth smoothing techniques have been proposed to solve this problem [12]. Bandwidth smoothing technique is to enhance the overall bandwidth efficiency according to the burst time and media size. The focus in this technique is to prefetch the media data to the decoder buffer ahead of bursts of media data. This technique has been first proposed for QoS-guaranteed networks, and can be classified into offline smoothing for stored media and online smoothing for live media [13]. However, this technique has the drawback which must be known concisely the information of the encoding rate and decoding rate in the buffer.

3. The proposed AMSS Mechanism

In the wireless mobile environment it is difficult for protocols such as RTP [2] and RTCP [1] to know packet loss and delay [8, 9] adaptively. This section describes the AMSS for adaptive multi-level streaming and figure 1 shows the AMSS system structure. The proposed AMSS measures the similarity using fuzzy value μ [14].

The proposed AMSS is performed as the following.

1) AMSS fetches media block objects for streaming from the media contents server.

2) AMSS compose of hierarchically the stream buckets.

3) AMSS analyses the fuzzy similarity for stream relevance.

4) AMSS performs the streaming service decision for the analyzed stream relevance metric.

5) AMSS communicate with RTP/RTSP server module

for the request of streaming media objects.
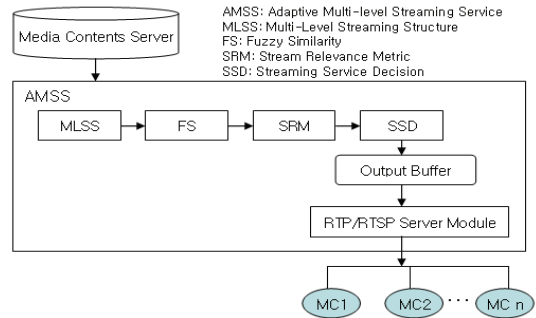
[Fig. 1] System Structure

3.1 MLSS

In wireless mobile networks, packets to be streamed in units of packet should be stored in continuous stream buckets. If data packets are not streamed adaptively from buckets then there occurs shortage of memory due to caching. Shortage of memory leads to congestion and delay which cause re-streaming and overhead [15].

In the proposed method data packets are stored into the temporary bucket during the initial streaming level for adaptive multi-level streaming. Streaming service in the proposed method is efficient when the number of data packet streams increases. In multi-level stream structure the determination of level depends on burst time and data packet size which influences the streaming service. The size of data packets and burst time are very important factors to determine the similarity in the proposed method. Let $[t_s, t_e]$ to denote the burst time interval t_s for multi-level adaptive streaming. t_s, and t_e are the start time and the end time of the burst time interval, respectively.

At first L-0 level is composed of bursts and the burst block B_h is formed for streaming level decision. The next upper level L-1 is composed of stream pairs with higher
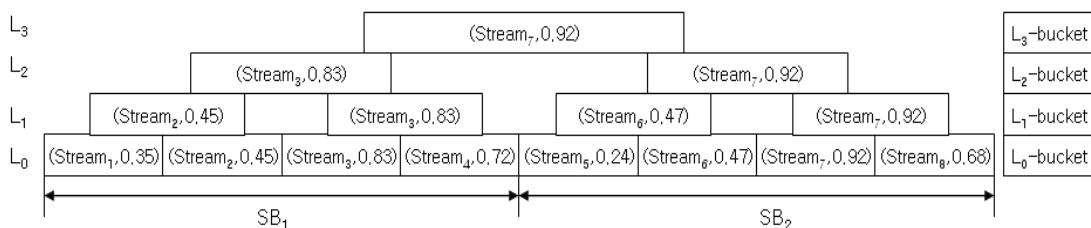
[Fig. 2] Multi-level Stream Structure
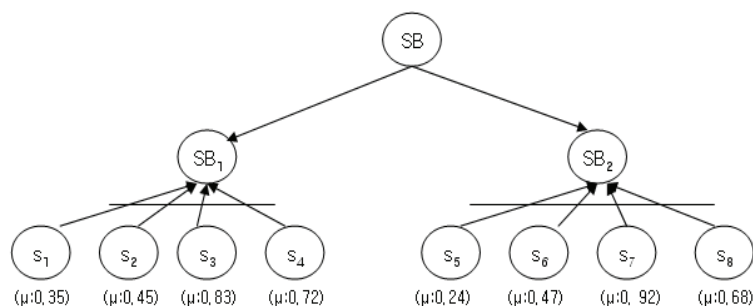
[Fig. 3] FS process

similarity metric and lower cost determined by average similarity cost of stream pairs. This process continues iteratively until the end of time interval or t_e^L. L=4, B_h=8, and t_s=8. In Figure 2, L-0 is the initial level and L-1, L-2, and L-3 represent level-1, level-2, and level-3, respectively. The input value of level-0 is stored in the cache block bucket first. Cache block buckets in the level store the stream fuzzy value μ. Generally L-0 is composed of packet streams with random the data packet size and burst time that does not decide the sequence of streaming and without the consideration of similarity.

Repeated use of similarity metric generates the higher levels. Finally L-3 has the highest similarity streams. In other words, L-3 has the most optimum streams with the least number of burst operations. L-3 streams moved from L-0 step by step. Each level maintains the fuzzy value μ of the latest level stream. If level L-k has an input of new streams, then the fuzzy value of L-k generates the new level L-(k+1) by using the similarity parameter μ and updates the end time of L-(k+1) or t_e^{k+1}.

3.2 FS

FS process of figure 3 determines the sequence of similarity between stream packets. The sequence of stream packet is performed by the similarity measurement.

This method is used to determine whether a stream packet relates with the other stream packets and to solve the problem of similarity duplication. For the stream bucket SB_i and any stream packet p_j in SB_i, let K denote the set of all child node of SB_i and N denote the number for stream packets in SB_i. Then FS for SB_i of stream packets is defined by the following.

Definition 1. FS=

$$\sum_{s_j \in child\ of\ parent\ SB_i} \{\frac{SB_i(s_j)}{N}\} \geq \mu - cut$$

3.3 SRM

The relevance metric to find optimum streams starts in stream buckets by using the fuzzy similarity for each stream. The relevance metric proposed in this paper is to minimize the packet loss due to congestion and conflict found in some routings and clients in wireless mobile environment and to minimize re-streaming due to disconnection and delay [7, 15].

For the measurement of similarity metric the search to the upper/lower levels is performed by the query in multi-level stream structure. Streams satisfying the fuzzy similarity is eligible for the upper level, has a small number of burst operations, and generates less packet loss due to congestion and delay.

Contrary to this case the lower layer streams, especially L-0 streams, are those without fuzzy similarity and thus requires a lot of burst operations and may suffer bandwidth limitation and delay. In order to reduce this kind of problem, the lower level streams should be described in more detail than the upper layer streams. Existing streaming methods, however, do not describe so detail as to reduce packet loss and re-streaming due to congestion and delay.

The proposed method tests similarity metrics of streams to minimize packet loss and re-streaming at each level. Similarity decision ensures stable streaming and improved performance. Similarity is checked as the following.

Definition 2. In order to measure the similarity of stream s_j in stream bucket SB_i at each level, the similarity of j-th stream data is defined as the following.

The similarity of stream s_j in stream bucket SB_i is $\{(s_a(L_1), \mid \mu_1 \mid), (s_a(L_2), \mid \mu_2 \mid), \ldots, ((s_a(L_j), \mid \mu_j \mid)\}$, where L_j is the j-th stream at level L.

After the similarity check for each stream at each level, cost for the similarity is measured. The total cost for streams at level L is determined by the cost for each stream and given as the following.

Definition 3. Similarity cost for the stream searched in SB_i at level L is $SR_{cost}SB_i(s_a) = \sum_{i=1}^{N} \omega_i CostSB_i(s_a)/N$, where N is the number of searched streams and ω_i is the weight for the stream s_i.

Similarity cost for a stream is a parameter for the improvement of performance. The smaller similarity cost shows the better performance. Relationship metric is measured for stream pairs as another similarity metric. The relationship metric SR(x) between two substream pair of $stream_1$ and $stream_2$ in a stream bucket at level L is defined as the following.

Definition 4. The relationship of two substreams, $stream_1$ and $stream_2$, of a stream bucket at level L is $SR(x) = \{\sum_k (|stream_{2k} - stream_{2k-1}|)^2 \times \mu \times \beta \times |L_d|)\}$, where μ is the similarity between streams, β is the weight for the similarity, and L_d is the number of stream pairs used in the calculation of distance metric.

After the measurement of similarity cost and relationship at each level, similarity is tested for stream pairs satisfying conditions for similarity. The stream with maximum μ is selected as the most appropriate stream. The similarity metric $HR_L(x)$, used to select the most appropriate stream, is defined as the following.

Definition 5. The similarity metric to select the most appropriate stream at level L is $HR_L(x) = \max\{pairs ((stream_1, \mu), (stream_2, \mu))\}$.

Thus the measurement of similarity metric helps to improve the performance of streaming.

3.4 SSD

Streaming service decision is the process to search the optimal stream packet in the bucket. Streams with higher similarity has higher priority and enters into the upper level. Streams entered into the upper level is less sensitive to the distortion of encoding rate and decoding rate. However, due to burst time and large data size, there exists distortion of encoding rate and decoding rate for the low level stream packet data whose level is not undetermined. This paper measures maximum similarity to solve this problem. At each level the streaming decision between stream packets, S_D, is defined as the following.

Definition 6. $S_D =$
$$\max\{\sum_{i=1}^{n}\sum_{j=1}^{m} \frac{1}{(1+|\mu S_i - \mu S_j|)} + SR_{cost}SB_i(S_a)\}$$
where SB_i is the i-th stream bucket.

Multi-level streaming structure decides whether to try to enter into the upper level by comparing with the predetermined threshold after the determination of maximum fuzzy similarity. In order to decide the stream in the multi-level streaming structure, threshold μ is classified as 3 groups of large burst for $\mu \leq 0.5$, middle burst for $0.6 \leq \mu \leq 0.7$, and short burst for $\mu \geq 0.8$. Fuzzy similarity is measured only if the relevance of streaming is not less than the threshold value. After this procedure multi-level structure reallocates levels according to the updated fuzzy similarity. The updated multi-level structure manages streaming stably by classifying streams into congestion and un-congestion. This reduces overhead due to disconnection and retransmission by reducing packet loss. Without decision the relevance of streaming in buckets, however, there occurs packet loss due to start delay and congestion. The stream packet, whose relevance of streaming is finally determined, is cached first in the cache due to short burst time and higher priority. The next caching is performed for the stream packet with the highest similarity. The cache continues streaming for stream packets only if the similarity is greater than the threshold. The reason for this is to reduce overhead for stream packets with similarity less than or equal to the threshold. After the caching of stream packets with similarity greater than threshold, the cache searches the

stream list in order of similarity. The following procedure shows the process to determine the streaming list according to the similarity.

Procedure streaming decision

SB={s_1, s_2, ..., s_n}

//Stream packets in buckets

$$\sum_{i=1}^{n} |SB_i| \geq \mu - cut$$

// Stream decision according to the fuzzy similarity

StreamListIndex=0

//Initialization for empty stream list for every stream packets in buckets

if (StreamListindex> $\sum_{i=1}^{n} |SB_i| \geq \mu - cut$) then {$S_D$=

max{ $\sum_{i=1}^{n} \sum_{j=1}^{m} \dfrac{1}{(1+|\mu S_i - \mu S_j|)}$ +SR$_{cost}$SB$_i$(S$_a$)}

MaxRelList=GetSreamDataSet(StreamList[StreamListIndex]);

//Generation of list for stream packets satisfying fuzzy similarity

(GetStreamDataSet(StreamList[StreamListIndex]), StreamListIndex+1);

//Call the next stream packet to decide the continuous streaming

    }

else if(StreamListindex< $\sum_{i=1}^{n} |SB_i| \geq \mu - cut$) then{

//Control of streaming decision for stream packets

Threshold= $\sum_{i=1}^{n} |SB_i| \geq \mu - cut$

//Decision of level according to the threshold

else {F$_{maxRel}$ $\cap$ $\sum_{i=1}^{n} |SB_i| \neq 0$

//Decide whether to continue or stop the streaming service

$$\sum_{i=1}^{n} |SB_i| \leq 0.5 - cut$$

//Stop the streaming service

    }

4. Simulation results

This paper used NS-2 simulator for the simulation evaluation [16]. In the simulation the following parameters are assumed: bit rate is 1.28Mbps, the packet size for the streaming is 512kbyte, the bandwidth of the link is 10/100Mbps, the average bandwidth of the link is 1.2Mbps, t_s for the stream is [1, 20s], and the weighting factor $\mu \geq 0.5$.

Table 1 is the performance parameters.

[Table 1] Performance Parameters

| Parameter	Value
Number of stream object	1,500
Stream Object Size	100MB-200MB
Encoding Rate	1.2Mbyte
Best-Fit Similarity	μ=0.8, μ=0.9
Packet Size	512Kbyte
Bandwidth	10/100Mbps
t_s	[1,20s]
Similarity Weight	$\mu \geq 0.6$

The simulation is performed for the proposed AMSS, TCP-based method, UDP-based method and VBM-based method. In the proposed method best multi-level stream structure is assumed since the initial L-0 influences heavily on the performance of the multi-stream structure. S_{number}=CB (Cache Block) is used to denote the size and the number of the caching blocks in the simulation. Simulation is performed 5 times for five packet data streams sized 500, 1000, 1500, 2000, and 2500. After the simulation, the relevance is analyzed considering stream level, the size of caching block, and the size of packet data. Simulation scenarios are as the following.

1) Simulation scenario 1: Simulation is performed with 2,500 packet data to find the number of packet overheads for the proposed AMSS and non-AMSS. Non-AMSS is methods without the use of similarity metric.

2) Simulation scenario 2: Simulation is performed to find the congestion ratio for the multi-level stream structure with the relevance measure μ of 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9.

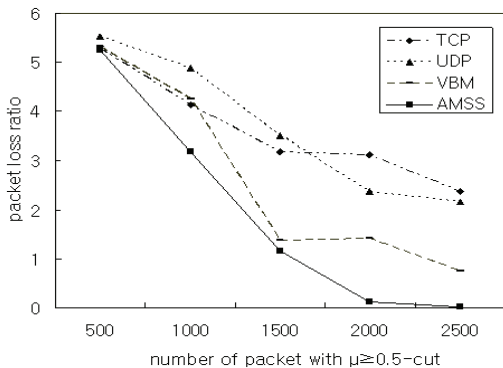3) Simulation scenario 3: Simulation is performed to

show the packet loss ratio with the number of packet of 500, 1000, 1500, and 2000.
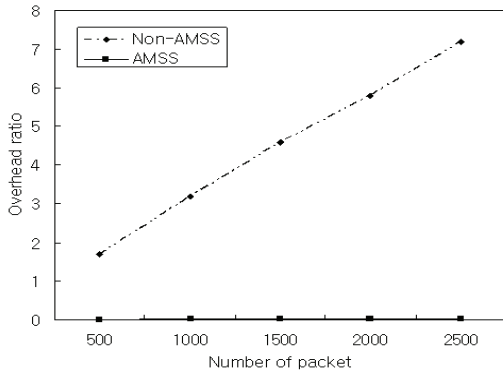
[Fig. 4] Overhead Ratio for AMSS and non-AMSS
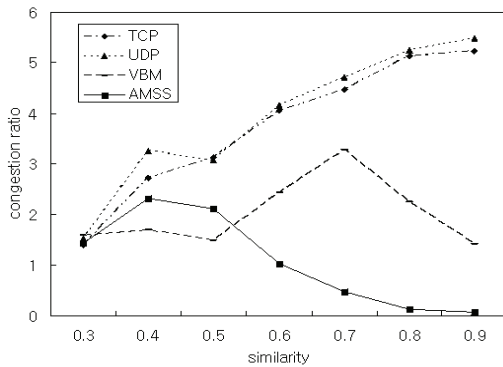
[Fig. 5] Congestion Ratio

[Fig. 6] Packet Loss Ratio

Figures 4, 5, and 6 show simulation results for scenarios 1, scenarios 2 and 3.

Figure 5 shows the simulation result with the relevance measure μ of 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9. Figure 6 shows the packet loss ratio with the number of packet of 500, 1000, 1500, and 2000. As shown in figure 5 and figure 6, the proposed method showed improvement of 10% in congestion control ratio and 8% in packet loss ratio compared with VBM method which is one of the best method. The proposed method showed average performance improvement of 10% in $\mu \geq 0.9$-cut and showed average performance improvement of 8% in the number of packet with 2,500. The more similarity means the better performance for congestion control and packet loss in the proposed AMSS method. Thus, it is shown that the proposed AMSS performs streaming adaptively and stably. Moreover the simulation shows that the proposed method performs congestion control more stably and reduces packet loss more than the other existing methods.

5. Conclusion

This paper proposed a AMSS mechanism to reduce the performance degradation due to overhead such as variable network bandwidth, mobility and limited resources of the wireless mobile network. The proposed method is capable of multi-level streaming service by the use of fuzzy similarity μ. For the improvement of streaming performance, next higher levels are organized successively to construct a multi-level stream structure by using the fuzzy similarity after the loading of initial stream buckets. An optimum stream is searched after the selection of pairs selected using the calculated cost and relevance metric by the fuzzy value μ. The searched optimum stream packet is selected for streaming. In particular, this paper measured S_D to decide the optimal streaming service in the multi-level structure. This decision enables the streaming performance to enhance when users request the streaming service. The NS-2 simulator is used for the simulation. The simulation result showed that the proposed AMSS method has better performance of 10% for congestion control ratio and 8% packet loss ratio than the other existing methods.

References

[1] C. Huitema, "Real-Time Control Protocol(RTCP) attribute in session Description Protocol(SDP)," IETF, RFC 3605, October, 2003.

[2] Audio-Video Transport Working Group, H. Schulzrinne, S. Casner, R. Frederick and V. Jacobsom, "RTP: A Transport Protocol for Real-Time Applications," IETF, RFC 1889, January 1996.

[3] A. Bulut and A. K. Singh, "Swat: Hierarchical Stream Summerization in Large Network," Proc. Int'l Conf. pp. 303-314. 2003.

[4] J. Jung, B. Krishnamurthy, and M. Rabinovich, "Flash Crowds and Denial of Service Attacks: Characterization and Implications for CDN's and WEb sites," in Proc, Int'l www Conf. 2002.

[5] C. Charu. A. J. Han, J. Wang, and S. Y. Phillips, "A Framework for Demand Classification of Evolving Data Streams," IEEE Trans. on Knowledge and Data Eng., vol. 18, no. 5, pp. 577-589, 2006.

[6] Z. Xiang, Q. Zhang, W. Zhu, and Y. Q. Zhang, "Peer-to-Peer Based Multimedia Distribution Services," IEEE Trans. on Multimedia, vol. 6, no. 2, pp. 343-354, 2004.

[7] B. Xie and W. Zeng, "Rate Distortion Optimized Dynamic Bitstream Switching for Scalable Video Streaming", In Proc. IEEE Int. Conf, Multimedia and Expo, Taipei, Taiwan, Jun, 2004.

[8] J. Widmer, R. Denda, and M. Murve, "A Survey on TCP-friendly Congestion Control", IEEE Network, vol. 15, no. 3, pp. 28-37, 2001.

[9] L. Cai, X. Shen, J. Pan, and J. W. Mark, "Performance Analysis of TCP-friendly AIMD Algorithms for Multimedia Apllications", IEEE Trans. Multimedia, Vol. 7, No. 2, pp. 339-355, 2005.

[10] N. R. Sastry and S. S. Lam, "CYRF : A Theory og Window-Based Uni-cast Congestion Control", IEEE/ACM Trans. Network, Vol. 13, N0. 2, pp. 330-342, 2005.

[11] M. Handly, S. Floyd, J. Padhye, and J. Widmer, "TCP-friendly Rate Control : Protocol Specification", in IETF RFC 3448, 2003.

[12] S. Sen, J. L. Rexford, J. K. Dey, J. F. Kurose, and D. F. Towsley, "On-line Smoothing of Variable B-t-Rate Streaming Video", IEEE Trans. Multimedia, Vol. 2, No. 1, pp. 37-48, 2000.

[13] T. Kim and M. H. Ammar, "Optimal Quality Adaptation for Scalable Encoded Video", IEEE J. Sel. Areas Commu, Vol. 23, No. 2, pp. 344-356, 2005.

[14] K. Atanassov, "Intuitionistic Fuzzy Sets," Fuzzy Sets Syst., vol.20, pp.87-96, 1986.

[15] P. Zhu, W. Zeng, and C. Li, "Joint Design of Source Rate Control and QoS-Aware Congestion Control for Video Streaming Over the Internet," IEEE Trans. on Multimedia, vol. 9, no.2, pp.366-376, 2007.

[16] NS-2 simulator, www.isi.edu/nanam/ns

Chong-Deuk Lee   [Regular member]

- Feb. 1983 : Chonbuk National University, Dept. of Computer Science, BS
- Feb. 1989 : Chonbuk National University, Dept. of Computer Science, MS
- Feb. 1998 : Chonbuk National University, Dept. of Computer Science, Ph.D
- Mar. 1992 ~ Feb. 2002 : Seonam University, Professor
- Mar. 2002 ~ Sept. 2010 : Chonbuk National University, Div. of Electronic Engineering, Professor

<Research Interests>
Wireless Mobile Networks, Wireless Sensor Networks, Wireless Mobile Ad hoc Networks, Ubiquitous Computing.