

자동차 환경내 안정적인 VoIP 시스템을 위한 네트워크 지터 추정 알고리즘

Network Jitter Estimation Algorithm for Robust VoIP System in Vehicle Environment

서 광 덕*
(Kwang-Duk Seo)

이 진 호**
(Jin-Ho Lee)

김 형 국***
(Hyoung-Gook Kim)

요 약

본 논문에서는 자동차 환경내 안정된 VoIP 통신시스템을 위한 새로운 네트워크 지터 추정 알고리즘을 제안한다. 제안된 알고리즘은 패킷간의 도착시간과 생성시간 차이를 계산하여 패킷이 겪은 현재 네트워크 환경을 추정하고, 추정된 네트워크 환경을 통해 네트워크 지터 추정에 사용될 네트워크 지터 분산 가중치를 조정된 후에, 조정된 지터 분산 가중치와 네트워크 지터의 평균과 분산을 계산하여 다음에 도착할 패킷의 네트워크 지터를 추정한다. 본 논문에서는 WiFi를 이용한 VoIP에서 수신단에 도착한 패킷에 대해 Delay와 Loss를 측정함으로써 제안된 방식의 우수성을 입증하였다.

Abstract

This paper proposes a novel network jitter estimation algorithm for robust VoIP communication system. The proposed method computes the current network environment mode using the differences of arrival time and generation time from sequential received packets. According to the current network environment mode, the jitter variance weights is adjusted to minimize the error for estimating the network jitter. The jitter average and variance are calculated by the autoregressive estimated algorithm, and then the network jitter is estimated by applying the jitter variance weights.

Key words : VoIP, spike detection, network jitter estimation, jitter buffer control

I. 서 론

통신 기술의 발전과 IT서비스의 확대로 음성·데이터, 방송·통신 등이 IP망으로 통합되는 All-IP가 진행되고 있다. 그중에서 기존 통신 사업자가 구축한 서킷망을 통한

음성 통화를 넘어서서 VoIP(Voice over Internet Protocol)에 대한 서비스의 요구가 확대되고 있으며[1], 이에 따른 VoIP의 품질 개선에 관한 연구가 진행되고 있다.

IP망에서의 음성 통화 서비스에서는 기존의 서킷망과 달리 네트워크 과부하, 패킷 지연 및 손실 등의

† 이 논문은 2010년도 광운대학교 교내 학술연구비 지원에 의해 연구되었음.

† 이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2011-0004311).

* 주저자 및 교신저자 : 광운대학교 전파공학과

** 공저자 : 광운대학교 전파공학과

*** 공저자 : 광운대학교 전파공학과

† 논문접수일 : 2011년 1월 13일

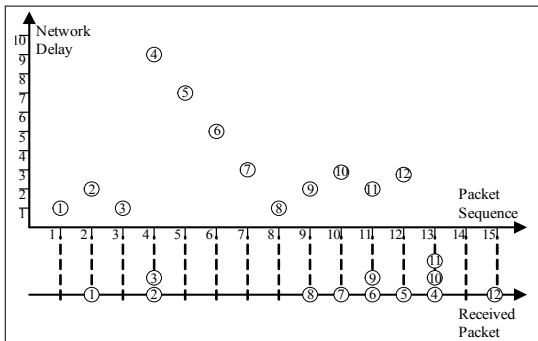
† 논문심사일 : 2011년 6월 13일

† 게재확정일 : 2011년 6월 13일

통신 서비스의 품질을 저해하는 요소들이 발생된다. 고품질의 VoIP가 제공되기 위해서는 우선적으로 VoIP의 수신단에서 발생된 네트워크 지터를 효율적으로 관리할 수 있는 적응적 지터 버퍼 제어(Adaptive Jitter Buffer Control)[2]가 필요하다. 이에 따라, 수신단에 도착한 패킷의 헤더 정보를 이용하여 패킷이 겪은 네트워크 지터를 예측하는 기술이 연구되어 오고 있다.

Jacobson[3]과 Ramjee[4]는 과거 네트워크 지터의 분산을 이용하여 Spike 네트워크 환경을 검출하고, 검출된 네트워크 환경에 따라 회귀적인 통계 수치를 조절하여 네트워크 지터를 추정하는 알고리즘을 제안하였으며, Fujimoto[5]는 회귀적인 통계 수치의 스무딩 변수를 발생하는 Loss과 Delay에 적응적으로 변화시키는 방법 등이 제시되었다. 또한 Narbutt[6]는 현재 네트워크 지터의 분산에 따라 스무딩 변수를 조정함으로써 정확한 네트워크 지터를 추정하였다. 하지만, 기존의 연구들은 송신단측에서 생성된 패킷의 순서대로 수신단에서 네트워크 지터를 추정해야만 정상적인 네트워크 지터를 추정할 수 있다는 단점이 존재한다.

<그림 1>은 실제 VoIP시스템에서 발생하는 네트워크 지연 증가 현상으로 생성된 패킷 시퀀스 번호와는 달리 패킷이 발생하는 네트워크 지연에 따라 순서가 뒤바뀌어 도착하는 상황을 나타낸다. 즉, ④/⑤패킷에서 네트워크에 이상이 생겨 네트워크 지연이 증가함으로 인해 ④/⑤패킷이 도착하기 이전에 ⑥/⑦/⑧/⑨패킷이 도착한다. 기존의 알고리즘에서는 순차적으로 다음에 도착할 패킷의 네트워크 지연을 추정하기 때문에 현재 네트워크 지터에 대한



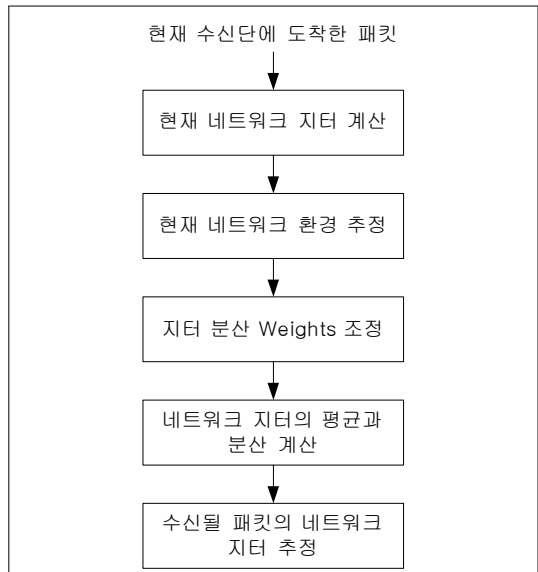
<그림 1> 네트워크 지연에 따른 패킷 도착의 예
<Fig. 1> Example of Packet Receive by the Network Delay

빠른 추정이 힘들고 변화하는 네트워크 지터와 패킷의 Delay과 Loss를 최소화하기에 어렵다.

본 논문에서는 자동차 환경내에 제공되는 VoIP 시스템에서 수신단의 네트워크 지터를 효율적으로 추정하는 알고리즘을 제안한다. 제안된 알고리즘에서는 수신단측에 순차적으로 도착하지 않는 패킷에 대해서도 네트워크 지터를 추정할 수 있고, 패킷의 손실과 지연 정도를 최소화할 수 있는 최적의 네트워크 지터 추정이 가능하다.

II. 네트워크 지터 추정 알고리즘

본 논문에서 제안한 네트워크 지터 추정 알고리즘은 <그림 2>와 같다.



<그림 2> 네트워크 지터 추정 알고리즘 블록도
<Fig. 2> Block Diagram of Network Jitter Estimation

제안한 방식에서는 현재 수신단에 도착한 패킷을 이용하여 현재 네트워크 지터를 계산하고, 패킷이 겪은 현재 네트워크 환경을 추정한다. 추정된 환경에 따라 네트워크 지터 추정에 사용될 지터 분산 가중치를 조정하고, 계산된 네트워크 평균과 분산을 이용하여 다음 수신될 패킷의 네트워크 지터를 추정한다.

1. 현재 네트워크 지터 계산

현재 수신단에 도착한 패킷과 이전 수신단에 도착한 패킷의 도착 및 생성시간을 비교하여, 패킷이 통과한 네트워크의 현재 지터 J_{c_i} 를 계산한다. 송신단에서 생성된 패킷은 변화하는 네트워크 지연을 겪어 수신단측에 도착하게 되며, 도착하는 패킷의 순서가 송신단에서 생성된 패킷 순서와 다를 수 있다. 이러한 점을 반영하여 패킷이 도착한 시간과 송신단에서 생성된 시간을 비교하여 현재 발생하는 네트워크 지터를 계산한다.

2. 현재 네트워크 환경 추정

본 논문에서는 변화하는 네트워크 환경에 따른 효과적인 네트워크 지터 추정을 위해, 네트워크 환경을 다음과 같이 3가지로 분류한다.

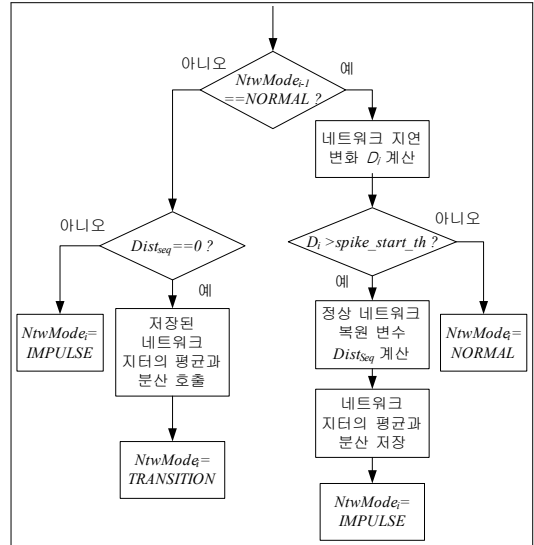
- 1) **NORMAL** : 정상적인 네트워크 환경을 가지는 구간으로, 네트워크 지터의 분산이 심하지 않고 일정한 패턴으로 네트워크 지터가 반복된다.
- 2) **IMPULSE** : 네트워크 지연이 갑자기 증가하는 Spike구간으로 증가 정도에 따라 버스트(Burst)의 형태를 나타내며, 불규칙적으로 네트워크 지터가 증가하거나 감소하게 된다.
- 3) **TRANSITION(천이)**: **NORMAL**와 **IMPULSE**의 경계점이 되는 구간으로, 네트워크 지터의 변화가 심하다.

위에서 분류한 네트워크 환경을 추정하기 위한 방법은 <그림 3>과 같다.

이전의 네트워크 환경이 **NORMAL**이면, **IMPULSE**를 검출하기 위한 네트워크 지연 변화 T_i^{mis} 를 (1)과 같이 계산한다.

$$T_i^{mis} = (a_i - a_{i-1}) + (t_i - t_{i-1}) \tag{1}$$

여기서 a_i 는 i 번째 패킷이 수신단에 도착한 시간이고, t_i 는 i 번째 패킷의 생성시간이다. (1)에서 계산된 네트워크 지연 변화가 비정상 네트워크 감지 문턱값(spk_spike_th)보다 작다면, 현재 네트워크 환경을 정상 네트워크 **NORMAL**로 판단한다.

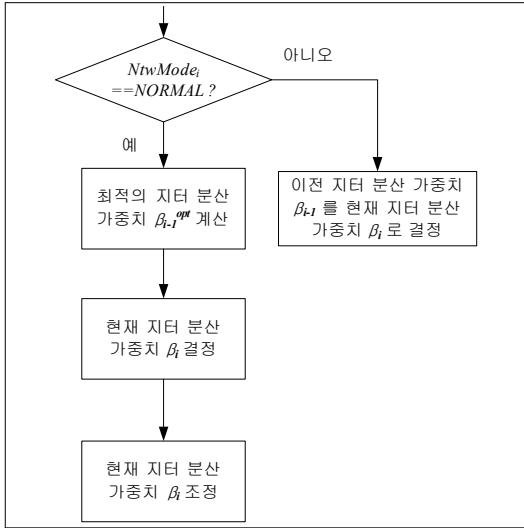


<그림 3> 네트워크 환경 추정 방법
(Fig. 3) Network Environment Mode Estimation

이와 반대로 네트워크 감지 문턱값보다 크다면, 현재 네트워크 환경을 비정상 네트워크 **IMPULSE**로 판단한다. **IMPULSE**의 끝점을 검출하기 위해 정상 네트워크 복원 변수를 계산하고, 네트워크 환경 변화시 빠른 적응력을 위해 이전에 계산된 네트워크 지터의 평균과 분산을 임시 저장한다. 이전의 네트워크 환경이 **NORMAL**이 아니고 **IMPULSE**인 경우에는, 위에서 계산된 정상 네트워크 복원변수를 이용하여 천이 네트워크를 감지한다. 천이 네트워크 감지를 위해, 현재 수신단에 도착한 패킷의 시퀀스 번호를 이용하여 정상 네트워크 복원 변수를 갱신하고, 갱신된 정상 네트워크 복원 변수가 0일 때 **TRANSITION**으로 판단한다. 그리고 이전에 임시 저장해둔 네트워크 지터의 평균과 분산을 호출한다. 이는 네트워크 환경이 **NORMAL**로 복원되었을 경우 이전에 존재하던 **NORMAL** 네트워크 환경의 통계적 수치를 적용시킴으로써 변화하는 네트워크 환경에 보다 빠르게 적응할 수 있다.

3. 지터 분산 가중치 조정

네트워크 지터를 효과적으로 추정하기 위해 추정된 네트워크 환경을 바탕으로 지터 분산 가중치



〈그림 4〉 네트워크 지터 분산 가중치 조절
(Fig. 4) Adjustment of the jitter variance weights

를 <그림 4>와 같이 조정한다.

현재 네트워크 환경이 NORMAL이 아니라면, 이전에 사용한 지터 분산 가중치 β_{i-1} 를 현재 지터 분산 가중치 β_i 로 지정한다. 반대로 현재 네트워크 환경이 NORMAL이라면, 이전에 조정되었어야 할 최적의 지터 분산 가중치를 계산한다. 최적의 지터 분산 가중치는 현재 네트워크 지터와 이전에 추정된 네트워크 지터간의 오차 e 를 최소로 갖는 값으로 다음 (2)과 같다.

$$e = 0 = J_{e_i} - J_{c_i} \quad (2)$$

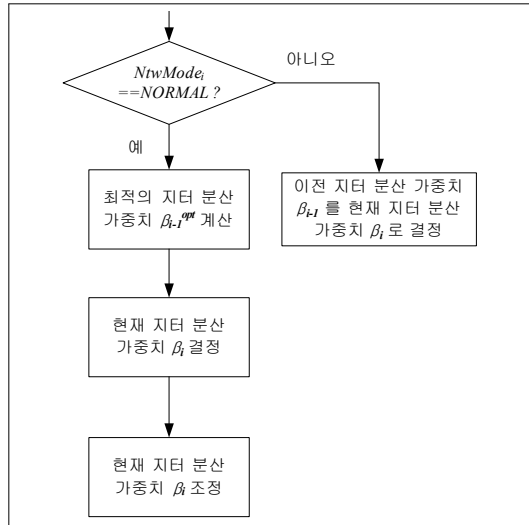
패킷이 겪는 네트워크 지터가 변하지 않는다면, 현재 지터 분산 가중치 β_i 는 오차를 0으로 하는 β 와 동일한 값을 가져야 한다. 하지만, 수신단에 도착하는 패킷마다 네트워크 지터가 변하기 때문에 현재 지터 분산 가중치 β_i 는 오차를 0으로 하는 β 와 동일한 값보다는 근접한 값을 지녀야 한다. 이러한 네트워크 지터가 변하는 것을 반영하여 β_i 를 오차를 0으로 하는 β 에 근접한 값으로 결정한다. 현재 지터 분산 가중치 β_i 를 결정한 후, 다음 (3)와 같이 현재 지터 분산 가중치 β_i 를 조절한다.

$$\begin{cases} \beta_i = \beta_{\max}, & \text{if}(\beta_{\max} < \beta_i) \\ \beta_i = \beta_{\min} + 0.2, & \text{elseif}(\beta_{\min} + 0.2 > \beta_i) \end{cases} \quad (3)$$

여기서 $\beta_{\min} = \max[\beta_{i-m} | m = 1, \dots, M]$ 이고, $\beta_{\max} = 2$, 그리고 $M=10$ 이다.

4. 네트워크 지터의 평균과 분산 계산

추정된 네트워크 환경에 따라 <그림 5>와 같이 네트워크 지터의 평균과 분산을 계산한다.



〈그림 5〉 네트워크 지터의 평균과 분산 계산
(Fig. 5) Computation of the jitter average and variance

현재 네트워크 환경이 TRANSITION이라면, 이전 네트워크 지터의 평균과 분산을 현재 네트워크 지터의 평균과 분산으로 지정하고 현재 네트워크 환경을 NORMAL로 판단한다. 이는 TRANSITION에서 발생할 수 있는 패킷 손실을 막고 정확한 네트워크 지터를 추정할 수 있게 한다. 반대로 현재 네트워크 환경이 TRANSITION이 아니라면, 현재 네트워크 지터의 평균 $avgJ_i$ 와 분산 $varJ_i$ 를 각각 (4), (5)과 같이 계산한다.

$$avgJ_i = \alpha \times avgJ_{i-1} + (1-\alpha)J_{c_i} \quad (4)$$

$$varJ_i = \alpha \times varJ_{i-1} + (1-\alpha) \times |avgJ_i - J_{c_i}| \quad (5)$$

여기서, α 는 스무딩 파라미터로 0.975를 가진다.

5. 수신될 패킷의 네트워크 지터 추정

위에서 계산된 네트워크 지터의 평균과 분산, 그리고 분산 가중치를 이용하여 (6)와 같이 다음에 수신될 패킷의 네트워크 지터 $J_{e_{i+1}}$ 를 추정한다.

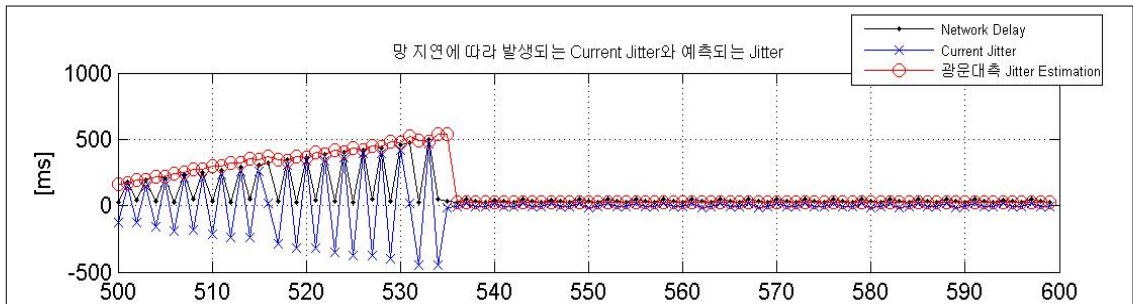
$$J_{e_{i+1}} = avg.J_i + \beta_i \times var.J_i \quad (6)$$

Ⅲ. 실험 결과 및 고찰

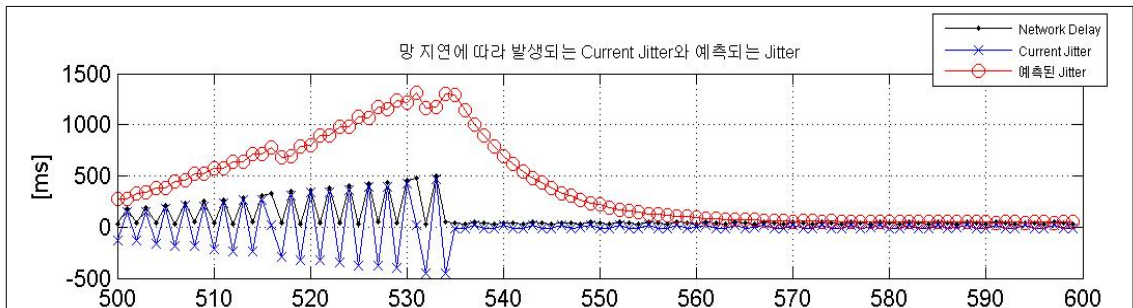
본 논문에서 제안된 네트워크 지터 추정 알고리즘의 성능을 측정하기 위해 자동차 환경의 WiFi망에서 채취된 패킷의 지연 정보를 이용하여 총 4개의 실험군을 구성하였으며, 버퍼링 Delay과 Loss에 대한 측정방식은 (7)에 나타나 있다.

$$\begin{cases} Delay : d = J_e - J_c, \text{ where } J_e - J_c \geq 0 \\ Loss : \text{Number of } i, \text{ where } J_e - J_c < 0 \end{cases} \quad (7)$$

<표 1>은 사용된 총 4개 실험군에 대한 네트워크 지연의 평균과 분산 정보를 나타내며, <표 2>은 실험군에 대한 기존 알고리즘[4]과 제안된 알고리즘의 성능 측정결과를 나타낸다. 제안된 알고리즘은 지연 성능 측정결과 실험군D에서 34.4ms로 가장 높은 성능을 보였으며 실험군B에서 46.1ms로 가장 낮은 성능을 보였다. 또한 손실성능 측정결과 실험군C에서 0.96%로 가장 높은 성능을 보였으며 실험군B에서 1.81%로 가장 낮은 성능을 보였다. 제안된 알고리즘이 기존 알고리즘보다 손실이 많은 원인은 Spike구간에서 증가되는 지터에 대해 지연의 최소화에 중점을 두었다. 이러한 최소의 지연은 지터 버퍼 컨트롤에 있어서 Spike구간에서 지터 버퍼의 크기를 유동적으로 조절함으로써 효율적인 지터 버퍼 제어가 가능할 수 있게 한다. 이에 <그림 6>과 <그림 7>은 실험군C에 대한 알고리즘별 네트워크 지터와 측정된 지연과 손실을 나타낸 것이다. 그림에서 제안된 알고리즘이 기존의 알고리즘보다



<그림 6> 제안된 알고리즘에 의한 네트워크 지터 추정 결과
<Fig. 6> Network Jitter Estimation of the Proposed Algorithm



<그림 7> 기존 알고리즘에 의한 네트워크 지터 추정 결과
<Fig. 7> Network Jitter Estimation of the Ramjee's Algorithm

Spike구간에서 변화되는 네트워크 지터에 빠르게 적응하여 지연을 최소화하는 것을 확인할 수 있다.

경과 비정상적인 네트워크 환경의 지터를 정확히 추정하여 지터 버퍼를 효율적으로 관리할 수 있다.

〈표 1〉 네트워크 지터 추정 실험군
 (Table 1) Test Database for Network Jitter Estimation

구분	패킷개수	네트워크 지연 평균	네트워크 지연 분산
실험군A	16567	49.29ms	26.02ms
실험군B	16348	42.17ms	57.75ms
실험군C	15000	48.79ms	31.97ms
실험군D	18246	47.17ms	34.77ms

〈표 2〉 네트워크 지터 추정 실험의 성능 측정 결과
 (Table 2) The Performance of the proposed Network Jitter Estimation Algorithm

구분	기존 알고리즘		제안된 알고리즘	
	Delay 측정결과	Loss 측정결과	Delay 측정결과	Loss 측정결과
실험군A	68.1ms	0.38%	34.4ms	1.17%
실험군B	78.0ms	0.44%	46.1ms	1.81%
실험군C	74.6ms	0.25%	36.9ms	0.96%
실험군D	69.5ms	0.34%	34.4ms	0.97%

V. 결 론

본 논문에서는 자동차내 VoIP시스템을 통한 음성 통화시 수신단측에서 발생되는 네트워크 지터를 효과적으로 추정하는 네트워크 지터 추정 알고리즘을 제안하였다. 제안된 알고리즘은 총 4개의 실험군에 대해 평균 Delay 37.93ms, 평균 Loss 1.23%를 보였다. 제안된 알고리즘은 정상적인 네트워크 환

참 고 문 헌

- [1] Batu Sat, Benjamin W.Wah, "Analyzing Voice Quality in Popular VoIP Applications," *2009 IEEE Computer Society, IEEE MultiMedia*, vol. 16, pp.46-59, Mar. 2009.
- [2] A. Kansal and A. Karandikar, "Adaptive Delay Estimation for Low Jitter Audio over Internet," *IEEE Global Telecommunications Conference*, vol. 4, pp.2591-2595, 2001.
- [3] V. Jacobson, "Congestion Avoidance and Control," *Proc. 1988 ACM SIGCOMM Conf., (Atlanta GA)*, pp.275-280, 1988.
- [4] R. Ramjee, J. Kurose, D. Towsley and H. Schulzrinne, "Adaptive Playout Mechanisms for Packetized Audio Applications in Wide-Area Networks," *INFOCOM '94, Networking for Global Communications., 13th, Proceedings IEEE*, vol.2, pp.680-688, 12-16, Jun. 1994.
- [5] K. Fujimoto, S. Ata. and M. Murata, "Adaptive Playout Buffer Algorithm for Enhancing Perceived Quality of Streaming Applications," *Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE*, 2451-2457, vol. 2, pp.17-21, Nov. 2002.
- [6] M. Narbutt and L. Merphy, "A New VoIP Adaptive Playout Algorithm," *Telecommunications Quality of Services: The Business of Success*, 2004, *Proceedings IEEE*, vol. 1, pp.99-103, Mar. 2004.

저자소개



서 광 덕 (Seo, Kwang-Duk)

2011년 3월 ~ 현 재 : 광운대학교 전파공학과 석사과정
2004년 3월 ~ 2011년 2월 : 광운대학교 전파공학과 학사



이 진 호 (Lee, Jin-Ho)

2010년 3월 ~ 현 재 : 광운대학교 전파공학과 석사과정
2003년 3월 ~ 2010년 2월 : 광운대학교 전파공학과 학사



김 형 국 (Kim, Hyung-Gook)

2007년 3월 ~ 현 재 : 광운대학교 전파공학과 부교수
2005년 4월 ~ 2007년 2월 : 삼성종합기술원 수석연구원
2002년 8월 ~ 2005년 3월 : 독일 베를린 공과대학교 Adjunct Professor
1999년 1월 ~ 2002년 7월 : 독일 Cortologic AG 책임연구원