무선 LAN 시스템에서의 SNR 오프셋을 이용한 링크 적응화

정회원 김 찬 홍*, 정 교 원**, 고 경 준*, 이 정 우*°

Link Adaptation with SNR Offset for Wireless LAN Systems

Chanhong Kim*, Kyowon Jeong**, Kyeongjun Ko*, Jungwoo Lee*° *Regular Members*

요 약

링크 적응 기법은 변하는 채널 조건에 맞는 최적의 MCS 레벨을 선택한다. 무선 LAN 시스템을 위한 다양한 링크 적응 알고리즘이 제안되었으나 802.11n과 같은 최근의 시스템에서 최적의 성능을 보장하지는 않는다. 본 논문에서는 수신 SNR과 전송 결과에 따라 얻어지는 오프셋 값을 이용한 새로운 링크 적응 알고리즘을 제안한다. 802.11n 시스템에서 모의실험을 하여 제안된 알고리즘과 잘 알려져 있는 ARF 및 일반적인 SNR기반 알고리즘과 그 성능을 비교해본다. 제안된 알고리즘은 PER에 제한이 있는 경우 시변채널에서 더 좋은 성능을 보인다.

Key Words : wireless LAN, SNR offset, link adaptation, 802.11n

ABSTRACT

Link Adaptation should select the best modulation and coding scheme (MCS) which gives the highest throughput as channel conditions vary. Several link adaptation algorithms for wireless local area network (WLAN) have been proposed but for the future WLAN systems such as 802.11n system, these algorithms do not guarantee the best performance. In this paper, we propose a new link adaptation algorithm in which an MCS level is chosen by the received SNR plus the offset value obtained from the transmission results. The performance of proposed algorithm is simulated by an IEEE 802.11n system. From the analysis, we conclude the proposed algorithm performs better than the well-known link adaptation algorithms such as auto rate fallback and general SNR-based techniques. Particularly, the proposed algorithm improves throughput when the packet error ratio (PER) is constrained for fast fading channels.

Ⅰ. Introduction

Wireless local area network (WLAN) technology has gained immense popularity in recent years. IEEE 802.11 technical group has proposed several standards for WLAN from 802.11a to g[1-4]. In order to prepare for increasing demands for multimedia services such as high-resolution TV and video on demand, a next generation high-speed WLAN standard, which is called 802.11n, has been announced[5]. The standard offers numerous modulation and coding scheme (MCS) levels to

※ 본 연구는 한국연구재단 일반연구자과제(2010-0013397), 중견연구자과제(2010-0027155) 및 서울기술혁신센터과제(JP091007, 0423-20090051) 지원과 서울대학교 뉴미디어통신공동연구소, BK21정보기술사업단 관리로 수행되었습니다.
* 서울대학교 전기·컴퓨터공학부 무선신호처리연구실(chkim@wspl.snu.ac.kr, smuff@wspl.snu.ac.kr, junglee@snu.ac.kr), (° : 교신저자)
** Department of Electrical and Computer Engineering, University of California at San Diego (k1jeong@ucsd.edu)
논문번호 : KICS2011-06-265, 접수일자 : 2011년 6월 20일, 최종논문접수일자 : 2011년 10월 16일

adapt transmission rate to varying channel condition, which is called link adaptation. Since the standard does not specify rate control algorithms, there is a freedom to use different type of link adaptation algorithms.

Link adaptation algorithms for WLAN have been developed by many researchers[6-12]. Most of them are based on channel statistics feedback or acknowledgement (ACK) message reception. These algorithms are inherently slow. Thus, when the link conditions degrade rapidly (e.g., when the user mobility is high), communication drop-out may occur. The short-term drop-out is normally handled by packet retransmission which increases packet delay significantly. This problem is tolerable for download applications, but is critical for streaming applications[10]. Therefore, a new link adaptation algorithm is needed in order to address the quality of service (QoS) issues for the future multimedia services.

In this paper, we propose a new link adaptation algorithm. It is similar to conventional SNR based link adaptation approaches in that a SNR-throughput lookup table is used. The difference is that it uses SNR offset to simplify the lookup table for different channel conditions. This paper is organized as follows. In Section II, conventional link adaptation algorithms to be compared are briefly reviewed. Next, the proposed link adaptation algorithm is presented in Section III. Simulation results and performance comparisons are described in Section IV. Finally, this paper makes its conclusion in Section V.

Ⅱ. Conventional Link Adaptation Algorithms

2.1 Auto rate fallback (ARF) link adaptation

In the 802.11 system, all successfully received data frames are acknowledged by sending an ACK frame to the sender. Hence, the number of consecutive ACK reception can be a measurement of the current channel condition. The algorithm is summarized as follows[11]:

1) It is assumed that changes in rates occur only by moving up or down one link rate.
2) Defines the three counting thresholds:
   - Step-up threshold N_i
   - Fallback threshold N_d
   - Probation threshold $N_p(<N_d)$
3) If consecutive ACK messages are received, the transmission rate is increased by one step.
4) If consecutive ACK messages are missed, the transmission rate is decreased by one step.
5) When the rate is increased, a probation state is used before committing to the new rate. If all of the next ACK messages are correctly received, the current rate is kept; otherwise, the transmission rate is reduced to the previous slower rate.

The optimal settings of the three thresholds are dependent on the link and the applications, but fixed thresholds are typically used in practice.

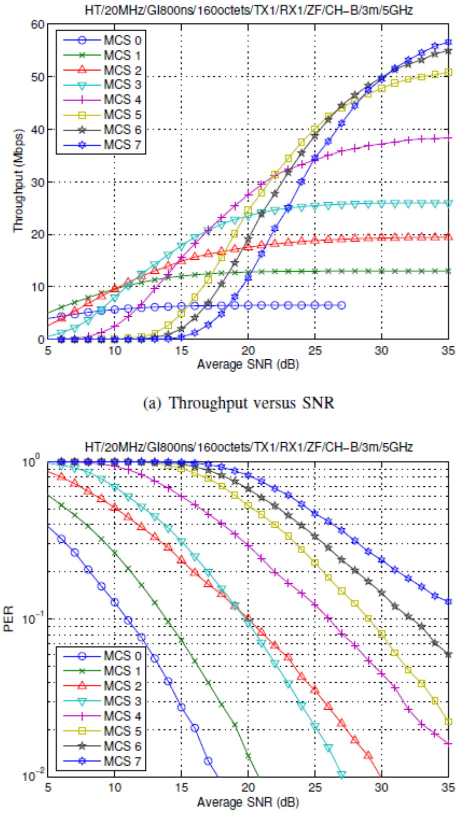
Although the ARF algorithm can be easily implemented by using only three counters, it has some disadvantages. Since the transmission rate moves up or down by only one step, the link adaptation speed is slow. The transition time also depends on initial MCS setting. Thus, its performance is poor in a fast fading channel. Moreover, when the short error bursts occur, the ARF algorithm causes throughput loss because moving to a higher rate takes much longer time than to a lower rate due to the need to collect meaningful channel statistics and to prevent oscillation. In addition to that, it is difficult and time-consuming to find out three optimal thresholds satisfying a PER constraint, which plays an important role of QoS.

2.2 SNR-based link adaptation

The SNR is directly related to the bit-error rate (BER) and to the channel capacity. Consequently, the SNR can be a good link quality indicator. If we know the current SNR and the SNR-throughput curves for each MCS level, the rate selection problem can be solved easily by switching to the rate which gives the highest throughput for the current SNR. This process can be implemented efficiently by using a SNR-throughput lookup table. If QoS needs to be satisfied, SNR-packet error ratio

(PER) curves can be used instead of SNR-throughput curves. For example, Fig. 1(a) and Fig. 1(b) show throughput and PER versus SNR curves for eight 802.11n SISO MCS levels, respectively. We can then construct lookup tables like Table 1 and 2 by means of Fig. 1. It is interesting to note that some MCS levels are not used, which is different from the ARF algorithm.

In spite of fast link adaptation speed, SNR-based techniques have not been applied in practice because it is difficult to get a reliable estimate of the SNR of a link. Since a lookup table depends on channel statistics, the rate corresponding to a given SNR in the lookup table may not be optimal for different channel conditions. There is a trade-off between throughput and PER. Even with this algorithm, it is also difficult to find the relationship of SNR range



(a) Throughput versus SNR

(b) PER versus SNR

Fig. 1. Throughput and PER versus SNR curves for 802.11n SISO MCS levels (TGn channel model "B" was used[12])

Table 1. SNR thresholds for link adaptation without PER constraint

MCS level	1	2	3	4	5	6	7
Data rate (Mbps)	13.0	19.5	26.0	39.0	52.0	58.5	65.0
SNR range (dB)	<10	10-12	12-17	17-22	22-27	27-31	>31

Table 2. SNR thresholds for link adaptation with PER constraint of 10%

MCS level	0	1	3	4	5	6
Data rate (Mbps)	6.5	13.0	26.0	39.0	52.0	58.5
SNR range (dB)	11-14	14-20	20-26	26-29	29-32	>32

and data rate which satisfies both PER constraint and maximum throughput at once.

Ⅲ. Proposed Link Adaptation Based on SNR Offset

We propose a new link adaptation algorithm which can overcome the disadvantage of the existing algorithms in Section II.

3.1 Algorithm summary

The SNR offset algorithm is summarized as follows:

1) Construct the SNR-throughput lookup table off-line for several different wireless channels.
2) The offset value is initialized as zero at first.
3) In the receiver, if a packet error occur, the offset value is increased by α.
4) If the current estimated SNR value plus the offset value exceeds the maximum SNR value of the lookup table, the offset value is not increased.
5) If the received packet does not have an error, the offset value is decreased by β.
6) If the current estimated SNR value plus the offset value is less than the minimum SNR value of the lookup table, the offset value is not decreased.
7) Add the updated offset value to the estimated SNR.
8) Choose an MCS level corresponding to the modified SNR value in the table.
9) The receiver requests the next packet with the selected MCS level.

The block diagram of the proposed algorithm is demonstrated in Fig. 2. If the SNR value can be fed

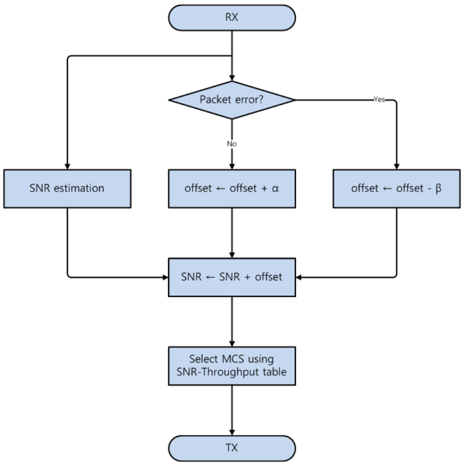


Fig. 2. Block diagram of the SNR offset link adaptation algorithm

back to the transmitter, 8) can be done in the transmitter.

3.2 Advantages of the SNR offset algorithm

Unlike the ARF algorithm in which the initial MCS level setting is important for shortening transition time, the proposed algorithm achieves short transition time and fast link adaptation speed. Because the algorithm basically uses a SNR-throughput lookup table for choosing a proper MCS level as in the conventional SNR- based technique. Consequently, it is more robust than the ARF algorithm in fast fading channels with short error bursts.

As mentioned above, the optimal SNR-throughput lookup table depends on the channel condition. Therefore, in conventional SNR-based algorithms, the selection of the optimal table is one of the most critical issues for the performance. However, in the proposed algorithm, we used a fixed SNR-throughput lookup table averaged over various channel conditions. In the following simulation results, it will be shown that the proposed algorithm performs well even if the used table is different from the optimal table. That is, unlike the conventional algorithms that depend on the accuracy of the estimated SNR and the lookup table, this algorithm can overcome SNR estimation error and mismatch between an MCS level and the SNR by using the SNR offset value. Furthermore, the offset value plays an important role in meeting the PER constraint. We found that the ratio of and is directly related to the PER performance from various simulations. For example, if PER is smaller than 0.1, we can set $\alpha=0.1$ and $\beta=1.0$ (dB). Although α and β should be found empirically as the counting thresholds of the ARF, it is comparably easier to meet the PER constraint in the proposed algorithm than in the ARF algorithm.

3.3 SNR estimation

Although the SNR offset can compensate estimation errors, the accuracy of SNR estimation is still an important issue. Because if we obtain more accurate SNR estimation, the transition time which it takes for an MCS level to be stable becomes much shorter. Thus, accurate SNR estimation can alleviate the throughput loss. One WLAN frame consists of several OFDM symbols. Generally, SNR is estimated by using the preamble of a frame. But when a user mobility is high, the channel may not stay fixed over a frame length any more. In order to address the fast fading problem, we propose two new SNR estimation methods.

Instead of using the preambles of each frame, the instantaneous SNR can be estimated by using the pilot and the null tones of the OFDM data symbols in a frame. The signal power of the nth frame can be denoted by

$$\widehat{\sigma_s^2}=\frac{1}{L_pL_d}\sum_{i,j}|p_{i,j}|^2, \tag{1}$$

where L_p is the number of pilot tones, and L_d is the number of data symbols in a frame. $p_{i,j}$ is the value of ith pilot tone of jth data symbol. Similarly, the noise power of the nth frame can be represented as

$$\widehat{\sigma_n^2}=\frac{1}{L_nL_d}\sum_{i,j}|n_{i,j}|^2, \tag{2}$$

where L_n is the number of null tones, and $n_{i,j}$ is the value of ith null tone of jth data symbol.

To average the instantaneous signal and noise power, two methods are proposed. In the first method, the signal power and the noise power of nth frame can be obtained by means of the weighted average as

$$\sigma_s^2(n) = \lambda\widehat{\sigma_s^2}(n) + (1-\lambda)\widehat{\sigma_s^2}(n-1), \quad (3)$$

$$\sigma_n^2(n) = \lambda\widehat{\sigma_n^2}(n) + (1-\lambda)\widehat{\sigma_n^2}(n-1), \quad (4)$$

where λ is a value near 1. Finally, the SNR of the nth frame can be estimated by

$$SNR(n) = \frac{\sigma_s^2(n)}{\sigma_n^2(n)}. \quad (5)$$

In the second method, we use the channel capacity formula in [13]. When the bandwidth W and SNR ρ is given, the channel capacity can be denoted by

$$C = W\log_2(1+\rho). \quad (6)$$

SNR can then be given by

$$\rho = 2^{\frac{C}{W}} - 1. \quad (7)$$

Since the SNR ρ is random, we can denote the expected value of the capacity by

$$E[C] = WE[\log_2(1+\rho)]. \quad (8)$$

If we assume ρ is an ergodic random variable, we have

$$\begin{aligned} E[C] &\approx W\frac{1}{N}\sum_{k=n-N+1}^{n}\log_2(1+\rho_k) \\ &= W\log_2\prod_{k=n-N+1}^{n}(1+\rho_k)^{\frac{1}{N}}, \end{aligned} \quad (9)$$

where ρ_k is given by

$$\rho_k = \frac{\sigma_s^2(k)}{\sigma_n^2(k)}. \quad (10)$$

Therefore, from (7) and (9), the average SNR can be expressed as

$$SNR(n) \approx \prod_{k=n-N+1}^{n}(1+\rho_k)^{\frac{1}{N}} - 1. \quad (11)$$

Ⅳ. Simulation Results

4.1 Simulation conditions

We implemented the physical layer of the IEEE 802.11n system[5] using MATLAB. In order to compare the performance of ARF, a general SNR-based technique, and the proposed SNR offset algorithm, we tested with only eight MCS levels of the 802.11n single-input single-output (SISO) environment. Simulation parameters are presented in Table 3. In addition, perfect synchronization and channel estimation are assumed. In the ARF method, the three counting thresholds are set to be $N_i = 8$, $N_d = 1$, and $N_p = 0$, respectively. These values are empirically found to be best in the given environment. In the SNR-based method and the proposed method, for each MCS level, the SNR-throughput curves were obtained by averaging throughput values for the 802.11n channel models (from A to E)[12]. 30,000 packets were used to generate the SNR-throughput lookup table, which was shown in Table 4. It was obtained from the SNR-throughput curves by choosing the MCS level

Table 3. Simulation parameters

Parameters	Value
PLCP format	High throughput mixed mode
Data length per frame	160bytes
Guard interval	800ns
Bandwidth	20MHz
Carrier frequency	5GHz
Used MCS levels	0 to 7
Used channel model	802.11 TGn A to E
Channel coding	Convolutional code
Number of antennas	Tx = 1, Rx = 1
Receiver	ZF

Table 4. SNR-throughput lookup table for 802.11n SISO MCS levels

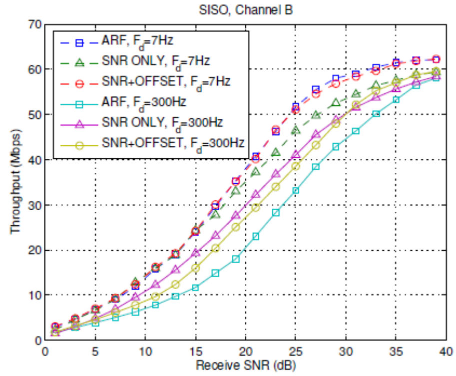
MCS level	0	1	3	4	5	6	7
Data rate (Mbps)	6.5	13.0	26.0	39.0	52.0	58.5	65.0
SNR range (dB)	<3	3-12	12-18	18-22	22-29	29-34	>34

generating maximum throughput value. Note that the MCS level 2 is excluded because throughput of other levels is higher than that of level 2, where SNR is estimated by the method of (5). In the proposed method, the SNR offset and are set to be 0.1 and 1 (dB), respectively, which makes PER below 0.1.

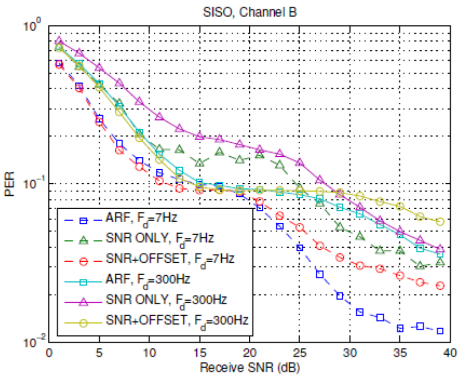
4.2 Performance comparisons

The throughput and the PER performances of the three link adaptation algorithms are shown in Fig. 3(a) and 3(b), respectively. From Fig. 3(a), we observe that the performance degradation of the ARF algorithm is significant when the Doppler frequency is increased from 7Hz to 300Hz. When the SNR is 25dB, the throughput loss is about 20 Mbps. However, the throughput loss is only about 12 Mbps in the proposed algorithm. The conventional SNR-only algorithm shows good throughput performance when Doppler frequency is high, but from Fig. 3(b), the PER is very high especially in the range the SNR-only algorithm shows good throughput performance. On the other hand, the proposed (SNR offset) algorithm not only shows good throughput performance, but also satisfies the PER requirement.

The performance of the proposed algorithm for various channels is demonstrated in Fig. 4. The proposed algorithm shows robust behavior for each channel. The use of the SNR offset value allows us to use only one SNR-throughput lookup table for different channel conditions. Finally, we compared the SNR estimation methods in Section III-3. In Fig. 5, the first estimation method corresponds to (5), and the second to (11). The second method showed higher performance. It should be noted that the

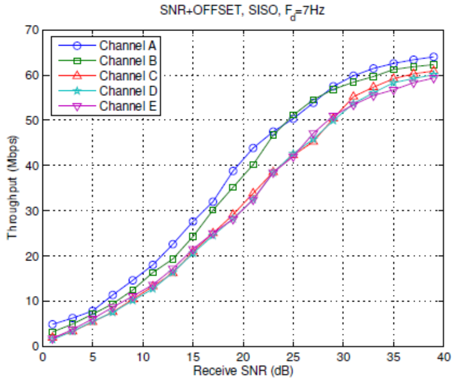


(a) Throughput versus receive SNR curves

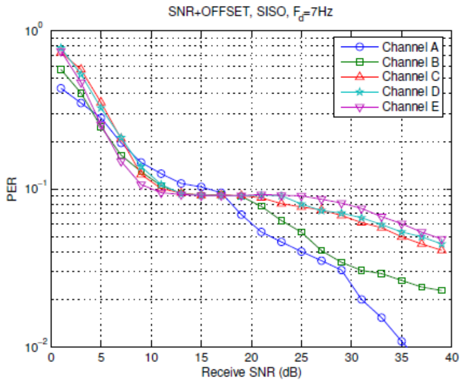


(b) PER versus receive SNR curves

Fig. 3. Performance comparison of the three link adaptation algorithms in slow and fast fading channels



(a) Throughput versus receive SNR curves



(b) PER versus receive SNR curves

Fig. 4. Performance of the proposed SNR offset algorithm with different channel models

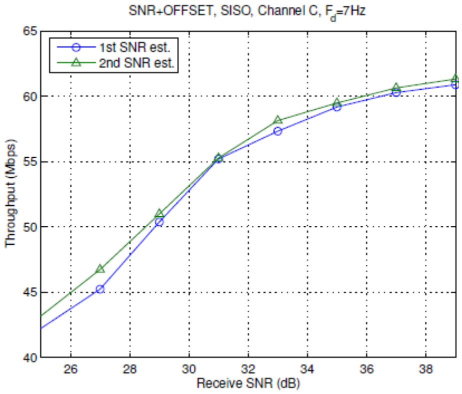


Fig. 5. Comparison of the two SNR estimation methods

performance can vary depending on the number of samples and the sampling duration.

Ⅴ. Conclusion

In this paper, we proposed the SNR offset link adaptation algorithm. Simulation results show that the algorithm performs better than the ARF algorithm and the general SNR-based technique especially in fast fading channels. The proposed algorithm has faster response in terms of link adaptation while it satisfies the PER requirement at the same time. It appears to be a good link adaptation technique for QoS sensitive applications. Since the number of available MCS levels gets larger as the number of transmit antennas increases, fast link adaptation will be more important in MIMO systems than in SISO systems. Although simulations are performed for the SISO case in this paper, the proposed algorithm can be easily extended to the MIMO case.

References

〔1〕 IEEE 802.11 “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”, Aug. 1999.

〔2〕 IEEE 802.11b “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4GHz Band”, Sep. 1999.

〔3〕 IEEE 802.11a “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer in the 5GHz Band”, Sep. 1999.

〔4〕 IEEE 802.11g “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 4: Further Higher Data Rate Extension in the 2.4GHz Band”, June 2003.

〔5〕 IEEE 802.11n/D1.0 Draft Amendment to STANDARD [FOR] Information Technology-Telecommunications and information exchange between systems-Local and Metropolitan networks-Specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Enhancements for Higher Throughput, Mar. 2006.

〔6〕 K. Balachandran, S. R. Kadaba, and S. Nanda, “Channel quality estimation and rate adaptation for cellular mobile radio”, IEEE Journal on Selected Areas in Communications, Vol.17, No. 7, pp.1244-1256, 1999.

〔7〕 J. Pavon and S. Choi, “Link adaptation strategy for IEEE 802.11 WLAN via received signal strength measurement”, IEEE International Conference on Communications, Vol.2, pp. 1108-1113, May 2003.

〔8〕 M. Heusse, F. Rousseau, G. B. Sabbatel, and A. Duda, “Performance anomaly of 802.11b”, INFOCOM, pp.836-843, Apr. 2003.

〔9〕 G. Holland, N. H. Vaidya, and P. Bahl, “A rate-adaptive MAC protocol for multi-hop wireless networks”, ACM MOBICOM, July 2001.

〔10〕 I. Haratcherev, J. Taal, K. Langendoen, R. Lagendijk and H. Sips, “Automatic IEEE 802.11 rate control for streaming applications”, Wireless Communications and Mobile Computing, Vol.5, pp.421-437, 2005.

〔11〕 A. J. Van der Vegt, “Auto rate fallback algorithm for the IEEE 802.11a standard”, Utrecht University, Tech. Rep., 2002.

〔12〕 V. Erceg et al,“TGn channel models”, IEEE 802.11 - 03/940r4, May 2004. http://www.802wirelessworld.com

〔13〕 B. Sklar, Digital communications, 2nd ed., Prentice Hall, 2001, pp.525-529.

김 찬 홍 (Chanhong Kim) 정회원

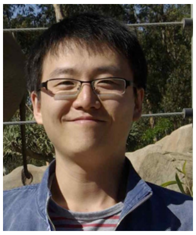
2004년 2월 서울대학교 전기공학부 졸업

2011년 2월 서울대학교 전기·컴퓨터공학부 박사

2011년 3월~현재 서울대학교 BK21 박사후연구원

<관심분야> 통신, 신호처리, random beamforming

정 교 원 (Kyowon Jeong) 정회원



2006년 2월 서울대학교 전기공학부 졸업

2007년 9월~현재 ECE, UCSD 박사과정

<관심분야> computational proteomics

고 경 준 (Kyeongjun Ko) 정회원

2006년 2월 서울대학교 전기공학부 졸업

2006년 3월~현재 서울대학교 전기·컴퓨터공학부 석박통합과정

<관심분야> 통신, 코드북 설계, limited feedback

이 정 우 (Jungwoo Lee) 정회원

1988년 2월 서울대학교 전자공학과 졸업

1990년 2월 프린스턴대학교 전기공학부 석사

1994년 6월 프린스턴대학교 전기공학부 박사

2002년 9월~현재 서울대학교 전기공학부 부교수

<관심분야> 무선통신, MIMO, 협력통신, 네트워크 코딩, 무선멀티미디어