Analysis of Rate-Compatible Punctured Serial Concatenated Convolutional Codes Based on SNR Evolution

Seung-Kyu Shin*, Dong-Joon Shin** *Regular Members*

ABSTRACT

The next generation mobile communication systems require error correcting schemes that can be adaptable to various code rates and lengths with negligible performance degradation. Serial concatenated convolutional codes can be a good candidate satisfying these requirements. In this paper, we propose new rate-compatible punctured serial concatenated convolutional code (RCPSCCC) which performs better than the RCPSCCC proposed by Chandran and Valenti in the sense of the rate compatibility. These codes are evaluated and analyzed by using computer simulation and SNR evolution technique. As their application, Type-II hybrid automatic repeat request (HARQ) schemes using both RCPSCCCs are constructed and new RCPSCCC is shown to have better throughput.

Key Words : SCCC, Rate-compatible codes, SNR evolution, Type-II HARQ

Ⅰ. Introduction

The next-generation mobile communication systems should support high data rate and asymmetric transmission as in the high speed downlink packet access(HSDPA) system[1]. Therefore, the frame error rate(FER) should be down to 10^{-4} ~10^{-5} at the target signal-to-noise ratio(SNR) range and the latency should be within 80ms with simple hardware structure. To be used for Type-II HARQ which is a popular technique to improve the throughput, the error-correcting codes should have the rate compatibility and show good performance for various codeword lengths.

The concept of rate-compatible codes was first presented in[2], where rate-compatible convolutional codes were obtained by adding rate-compatibility restriction to the puncturing rule. This restriction on puncturing requires that the rates are organized in the way that all the punctured bits of a lower-rate code should also be chosen as the punctured bits of all the higher-rate codes. The concept of rate-compatible codes was extended to Turbo codes, called rate compatible turbo codes (RCTCs)[3].

Turbo codes, also called parallel concatenated convolutional codes(PCCCs), have an error floor in the region of moderate-to-high SNRs due to low weight codewords[4]. Especially, at high code rate and short code length, the effect of error floor becomes more serious, which is an obstacle to the reliable packet transmission. On the other hand, serial concatenated convolutional codes (SCCCs) can provide better FER performance than Turbo codes in the region of moderate-to-high SNRs because SCCC shows little error floor[5]. Therefore, SCCC can be better suited for reliable packet transmission[5].

Chandran and Valenti[6] compared and analyzed RCTC and rate compatible punctured serial concatenated convolutional code(RCPSCCC). Although RCTC has error floor problem, it has better rate-compatibility than RCPSCCC[6].

Turbo codes are usually analyzed by SNR evo-

※ 본 연구는 정보통신부 ITRC 과제 지원 및 관리로 수행되었습니다.
* 서울통신기술(seungkyu.shin@samsung.com), ** 한양대학교 전자통신컴퓨터 공학부(djshin@hanyang.ac.kr), 교신저자
논문번호 : KICS2005-10-430, 접수일자 : 2005년 10월 30일, 최종논문접수일자 : 2006년 3월 15일

lution to obtain the threshold value [8]. In [9], the SNR evolution technique was used to explain many mysteries of Turbo code and SCCC.

In this paper, we propose new RCPSCCC which shows better performance than the RCPSCCC proposed in[6]. To obtain new RCPSCCC, good puncturing patterns are obtained through simulation. We also analyze new RCPSCCC by using SNR evolution technique. Furthermore, by employing RCPSCCC in Type-II HARQ based on incremental redundancy(IR) retransmission, we can obtain better throughput than Type-II HARQ using the existing RCPSCCC given in [6].

Ⅱ. Overview of Sccc

The structure of SCCC encoder is shown in Fig. 1, which consists of outer and inner convolutional encoders with interleaver between them. The information bit sequence U_O is encoded by the outer encoder. Then the output codeword C_O is interleaved and this interleaved sequence U_I is encoded by the inner encoder. The output C_I of inner code is transmitted through the channel.

SCCCs with non-systematic convolutional(NSC) code as the outer code and recursive systematic convolutional(RSC) code as the inner code are known to perform better than SCCCs with RSC codes as the outer and inner codes[5]. Based on this, we construct an SCCC with rate-1/2 NSC code as the outer code and rate-2/3 RSC code as the inner code and compare it with the SCCC

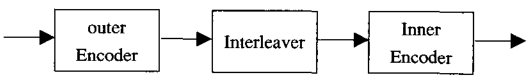


Fig. 1. Structure of SCCC encoder

Table 1. Generating Matrices of Convolutional Codes

Code	G(D)
Rate-1/2 NSC	$[1+D+D^2 \;\; 1+D^2]$
Rate-1/2 RSC	$\left[1 \;\; \frac{1+D^2}{1+D+D^2}\right]$
Rate-2/3 RSC	$\begin{bmatrix} 1 & 0 & \frac{1+D^2}{1+D+D^2} \\ 0 & 1 & \frac{1+D}{1+D+D^2} \end{bmatrix}$

Table 2. Code Structure

Code	Outer code	Inner code
SCCC1	Rate-1/2 NSC	Rate-2/3 RSC
SCCC2	Rate-1/2 RSC	Rate-1/2 RSC

with rate-1/2 RSC codes as the outer and inner code, with puncturer between them[6]. The convolutional codes used to construct these SCCCs are listed in Table 1. We denote new SCCC bySCCC1 and the code proposed in [6] by SCCC2 as in Table 2.

Ⅲ. Good Puncturing Pattrens and Construction of Rcpsccc

The puncturing rules can be classified into two types. The first type is to puncture only the parity bits. The second type is to puncture both the information and parity bits. When RSC codes are used as in SCCC2, the first type shows better performance than the second type. However, since an NSC code is used as the outer code for SCCC1, we should compare two types of puncturing rules to obtain good puncturing patterns.

Since the overall code rate of SCCC1 is 1/3, we choose the rates from 1/3 to 4/5 as the target code rates of RCPSCCC and design the corresponding rate-compatible puncturing patterns. For this, we consider the following three cases to obtain good puncturing patterns. Note that to design good rate-compatible puncturing patterns for the SCCC1, the puncturing is limited to the inner codeword bits.

Case 1: Puncturing only the parity bits.
Case 2: Puncturing only the information bits.
Case 3: Puncturing both the information and parity bits.

Candidate puncturing patterns for each case are listed in Table 3. Good puncturing patterns are selected through simulations. In our simulations, we assume that the channel is additive white Gaussian noise(AWGN) channel. Max-Log MAP is used in decoding since it is practically more widely used. We use puncturing period P=8 and

Table 3. Puncturing Patterns(Octal)

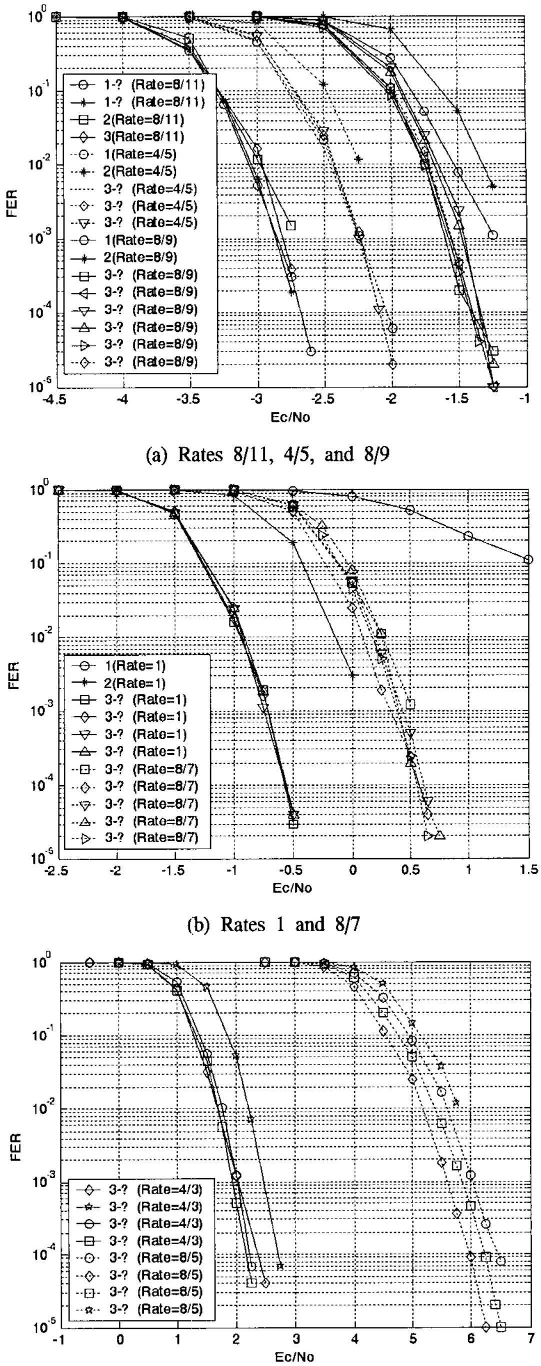
Inner Code Rate	Case	Puncturing patterns for inner encoder outputs
8/11	1	① [377; 377; 273] ② [377; 377; 356]
	2	[357; 376; 377]
	3	[357; 377; 376]
4/5	1	[377; 377; 252]
	2	[356; 167; 377]
	3	① [377; 375; 253] ② [376; 177; 273] ③ [376; 376; 356]
8/9	1	[377; 377; 210]
	2	[352; 127; 377]
	3	① [273; 375; 253] ② [377; 335; 252] ③ [356; 167; 273] ④ [356; 356; 356] ⑤ [376; 177; 252] ⑥ [337; 375; 252]
1	1	[377; 377; 000]
	2	[252; 123; 377]
	3	① [273; 335; 252] ② [377; 125; 252] ③ [356; 167; 252] ④ [356; 356; 314]
8/7	3	① [356; 167; 210] ② [273; 325; 250] ③ [356; 125; 252] ④ [352; 127; 252] ⑤ [354; 354; 354]
4/3	3	① [253; 325; 210] ② [252; 125; 252] ③ [352; 127; 210] ④ [354; 354; 210]
8/5	3	① [253; 123; 210] ② [252; 125; 210] ③ [314; 314; 210] ④ [252; 123; 222]

Table 4. Good Puncturing Patterns(Octal)

Rate	8/5	4/3	8/7	1
Punturing Patterns	[252; 125; 210]	[352; 127; 210]	[352; 127; 252]	[356; 167; 252]
Rate	8/9	4/5	8/11	2/3
Punturing Patterns	[376; 177; 252]	[376; 177; 273]	[377; 377; 273]	[377; 377; 377]

1000 information bits. For the purpose of comparison, the 8-state Turbo code given in [7] is also considered, which is known to have the complexity nearly close to that of 4-state SCCC. To construct an RCPSCCC from SCCC1, we choose one from the puncturing patterns at each rate in Table 3. Note that the inner code rate can be greater than 1 and the overall code rate is obtained by multiplying the outer code rate 1/2 to the inner code rate.

In Fig. 2 (a), at rate 8/11, 1-① is similar to 1-②.



(a) Rates 8/11, 4/5, and 8/9

(b) Rates 1 and 8/7

(c) Rates 4/3 and 8/5

Fig. 2. FER of various puncturing patterns for the inner codeword

3-② has the best performance at rate 4/5 and 3-① shows the best performance at rate 8/9. Because 3-① shows error floor, we select 3-⑤. In Fig. 2 (b), at rate 1, all patterns show nearly

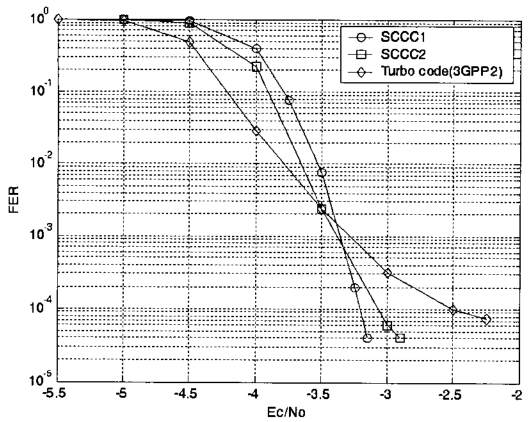


Fig. 3. FER of SCCC1, SCCC2 and Turbo code at rate 1/3

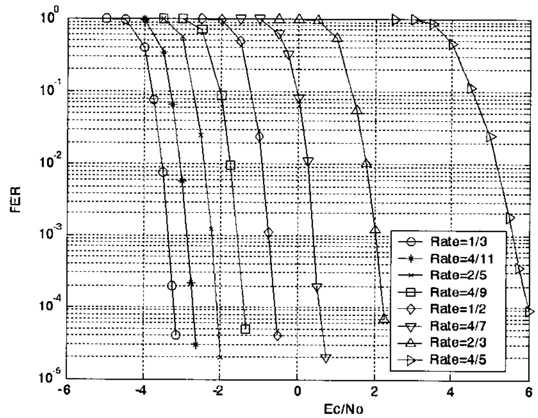


Fig. 4. FER of SCCC1 for the various rates

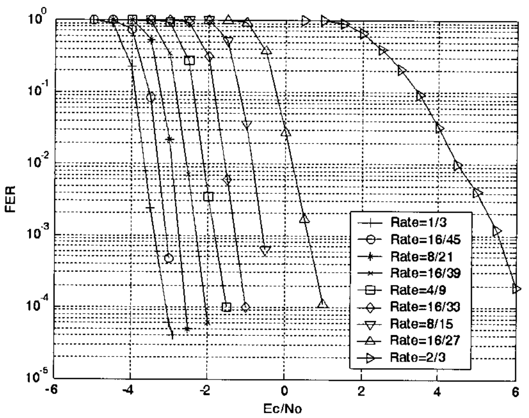


Fig. 5. FER of SCCC2 for the various rates

the same performance except 1 and 2. 3-② shows slightly better performance than the others at rate 8/7. Due to the rate-compatibility, we choose 3-④. In Fig. 2 (c), 3-③ is similar to 3-④ at rate 4/3. At rate 8/5, 3-② has the best performance.

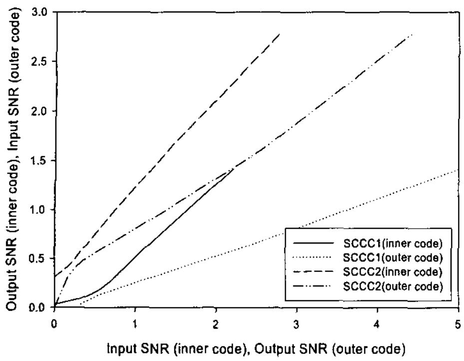
Based on these results, good rate-compatible puncturing patterns for target rates are selected and listed in Table 4.

Fig. 3 compares SCCC1, SCCC2, and Turbo code for the mother code rate 1/3. At SNR below -3.5dB, Turbo code has the best performance. However, at SNR above -3.5dB, Turbo code shows error floor, and SCCC1 and SCCC2 outperform Turbo code. Figs 4 and 5 show the FER performances of SCCC1 using puncturing patterns given in Table 4 and SCCC2 in [6], respectively. SCCC1 shows similar performance as SCCC2 at low code rates. Note that if the code rates 1/3, 4/9, and 2/3 are used to form a Type-II HARQ, at the rate 1/3, SCCC1 performs better for SNR below -3.5dB, SCCC2 slightly outperforms SCCC1 at the rate 4/9, and SCCC1 becomes much better than SCCC2 at the highest code rate 2/3, that is a crucial factor for the throughput.

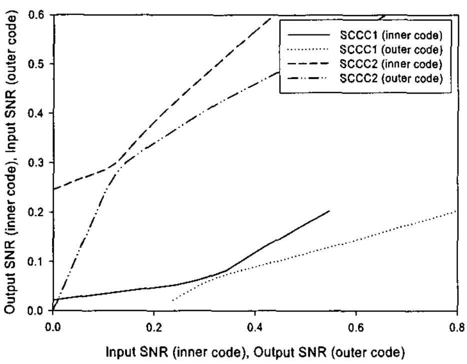
Ⅳ. Snr Evolution Analysis

For any serial or parallel concatenated convolutional code, we can compute the input and output SNRs at each iteration for each component decoder, using actual density evolution or the symmetric Gaussian approximation. The actual density evolution is calculated based on the collected LLRs at each iterative decoding step. If the collected LLRs can be approximated by a Gaussian density function, then its statistics depend only on two parameters, its mean μ and variance σ^2. Then, SNR for this random variable can be defined as $SNR=\mu/2$. If the collected LLRs are assumed both Gaussian and symmetric, then $\sigma^2=2\mu$ and $SNR=\mu/2$. The SNR evolution analysis is to plot the output SNR versus the input SNR for one component decoder, and the input SNR versus the output SNR for the other component decoder. If these two curves do not intersect, then the iterative decoder converges.

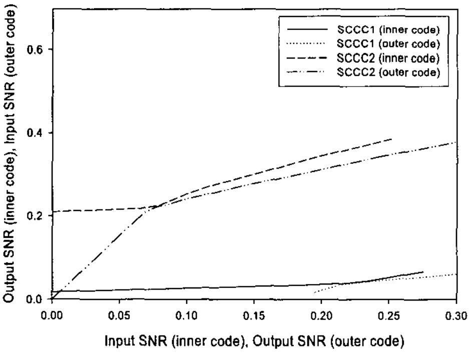
We compare SCCC1 and SCCC2 at rates 1/3, 4/9, and 2/3. In Fig. 6, it is verified that both SCCC1 and SCCC2 touch the iteration tunnel at



(a) Eb/N0=0.5dB



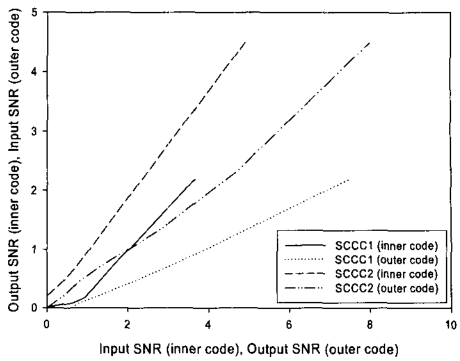
(b) Eb/N0=0.2dB



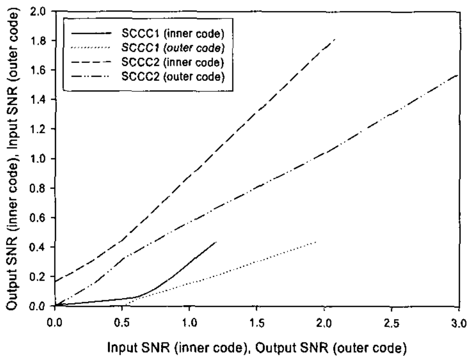
(c) Eb/N0=0dB

Fig. 6. SNR evolution at rate 1/3

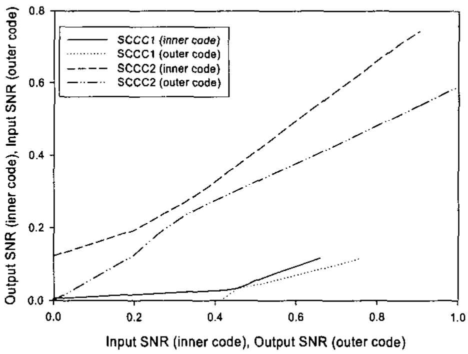
E_b/N_0=0 dB. Therefore, SCCC1 performs similarly as SCCC2 at rate 1/3 as given in Figs 4 and 5. In Fig. 7, SCCC1 touches the iteration tunnel at E_b/N_0=0.5 but SCCC2 shows iteration tunnel. Thus SCCC2 shows slightly better threshold(performance) than SCCC1 at rate 4/9 as given in Figs 4 and 5. Contrary to the case when rate 4/9, SCCC2 touches the iteration tunnel at inner input E_b/N_0=8dB. Thus, SCCC1 shows better performance than SCCC2 at rate 2/3 as given in Figs 4 and 5.



(a) Eb/N0=1.0dB



(b) Eb/N0=0.8dB



(c) Eb/N0=0.5dB

Fig. 7. SNR evolution at rate 4/9

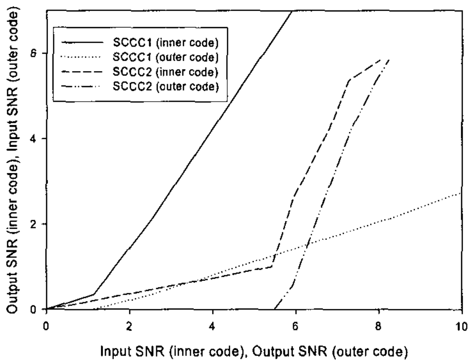


Fig. 8. SNR evolution at rate 2/3 for Eb/N0=4dB

V. Performance Of Type-Ii Harq

Type-II HARQ schemes are constructed by using SCCC1 and SCCC2, respectively. For the simulation, 1000 information bits, BPSK modulation, random interleaver, 8 iterations and AWGN channel are also assumed. It is assumed that the noiseless feedback link is available so that the receiver can reliably inform the transmitter of the decoding result. The round trip delay is not considered. The CRC code of length 1000 with the generator polynomial $x^{16}+x^{15}+x^2+1$ is used. SCCC1 with puncturing patterns for the rates 1/3, 4/9, 2/3 in Table 4 and SCCC2 with rate-compatible puncturing patterns given in [6] are used for the simulation. Fig. 9 shows that at SNR below -2dB, HARQ-SCCC2 shows slightly better performance than HARQ-SCCC1 but, for the SNR range of 0.5 and 4 dB, HARQ-SCCC1 outperforms HARQ-SCCC2.

VI. Conclusions

In this paper, we have proposed good rate-compatible puncturing patterns for new SCCC to construct RCPSCCC. For this code, by only puncturing inner codeword bits, RCPSCCC could achieve higher rates and better performance. The performance is analyzed both by simulation and SNR evolution technique. As its application, Type-II HARQ scheme using new SCCC is constructed and compared with the existing scheme to show that new RCPSCCC has better throughput. For the future research, various types of RCPSCCC suitable for various applications can be constructed and analyzed.

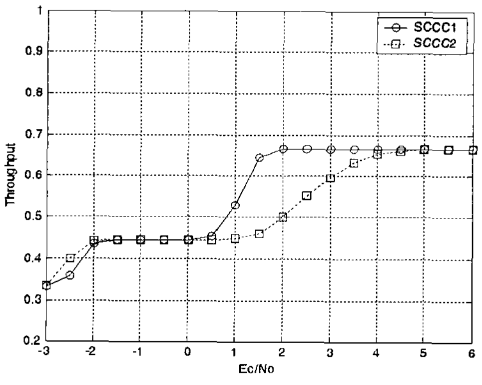


Fig. 9. Throughput of Type-II HARQ schemes using SCCC1 and SCCC2

REFERENCES

〔1〕 3rd Generation Partnership Project(3GPP), "High speed downlink packet access," *3GPP TR 25.855*, June 2001.

〔2〕 J. Hagenauer, "Rate-compatible punctured convolutional codes(RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. 36, pp. 389-400, Apr. 1988.

〔3〕 A. S. Barbulescu and S. S. Pietrobon, "Rate compatible Turbo-codes," *Electron.Lett.*, vol. 31, pp. 535-536, Mar. 1995.

〔4〕 L. C. Perez, J. Seghers, and D. J. Costello, "A distance spectrum interpretation of Turbo codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1698-1709, Nov. 1996.

〔5〕 S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design and iterative decoding," *IEEE Trans. Inform. Theory*, vol. 44, pp. 909-926, May 1998.

〔6〕 N. Chandran and M. C. Valenti, "Hybrid ARQ using serial concatenated convolutional codes over fading channels," *VTC'01 Spring*, vol. 2, pp. 1410-1414, May 2001.

〔7〕 H. Kim and G. L. Stüber, "Rate compatible punctured turbo coding for W-CDMA," in proc. *IEEE ICPWC'00*, pp. 143-147, Dec. 2000.

〔8〕 D. Divsalar, S. Dolinear, and F. Pollara, "Iterative turbo decoder analysis based on density evolution," *TMO Progress Reps.*, JPL, Feb. 2001.

〔9〕 H. E. Gamal and J. A. R. Hammons, "Iterative turbo decoder using the Gaussian approximation," *IEEE Trans. Inform. Theory*, Vol. 47, pp. 671-686, Feb. 2001.

신 승 규 (Seung-Kyu Shin) 정회원

2003년 2월 울산대학교 전자공학과 공학사

2005년 2월 한양대학교 전자통신전파공학 석사

2005년 3월~현재 서울통신기술 통신시스템연구소 연구원

<관심분야> 디지털통신, 오류정정부호

신 동 준 (Dong-Joon Shin) 정회원

1990년 2월 서울대학교 전자공학 공학사

1991년12월 Northwestern University 전기공학 석사

1998년 12월 University of Southern California 전기공학과 공학박사

1999년 1월~1999년 4월 Research Associate(USC)

1999년 4월~2000년 8월 Hughes Network Systems, MTS

2000년 9월~현재 한양대학교 전자전기컴퓨터공학부 조교수

<관심분야> 디지털통신, 오류정정부호, 이산수학, 시퀀스