퍼지 및 지능시스템학회 논문지 2001, Vol. 11, No. 6, pp. 538-542

DNA 코딩과 진화연산을 이용한 함수의 최적점 탐색방법

Global Optimum Searching Technique Using DNA Coding and Evolutionary Computing

백동화 · 강환일 · 김갑일 · 한승수
Dong-Hwa Peak, Hwan Il Kang, Kab Il Kim, and Seung-Soo Han

디지털 미디어 연구센터
명지대학교 전기정보제어공학부

요 약

DNA computing은 Adleman의 실험 이후에 많은 여러 가지의 최적화 문제에 적용되어 왔다. DNA computing의 장점은 스트링의 길이가 가변적이고 4가지 염기를 이용하기 때문에 복잡한 문제에서 전역 최적점을 찾는데 기존의 다른 방법보다는 효율적이라는 것이다.
본 논문에서는 이진 스트링의 개체 집단 위에서 모의진화를 일으켜 효율적으로 최적 해를 탐색하는 GA(Genetic Algorithms)와, 생체 분자인 DNA를 계산의 도구 및 정보 저장도구로 사용하며, A(Adenine), C(Cytosine), G(Guanine), T(Thymine)등의 4가지 염기를 사용하는 DNA 코딩 방법을 이용하여 multi-modal 함수의 전역 최적점을 탐색하는 문제에서의 각각의 성능을 조사하였다. Selection, crossover, mutation 등의 GA연산자를 DNA 코딩에 동일하게 적용하였으며 최적의 해를 탐색하는데 걸리는 시간과 찾아낸 최적해의 값을 평가하였다.

Abstract

DNA computing has been applied to the problem of getting an optimal solution since Adleman's experiment. DNA computing uses strings with various length and four-type bases that makes more useful for finding a global optimal solutions of the complex multi-modal problems.
This paper presents DNA coding method for finding optimal solution of the multi-modal function and compares the efficiency of this method with the genetic algorithms (GA). GA searches effectively an optimal solution via the artificial evolution of individual group of binary string and DNA coding method uses DNA molecules and four-type bases denoted by the A(Ademine), C(Cytosine), G(Guanine) and T(Thymine). The selection, crossover, mutation operators are applied to both DNA coding algorithm and genetic algorithms and the comparison has been performed. The results show that the DNA based algorithm performs better than GA.

Key words : DNA computing; GA; Optimization

1. 서 론

최근 들어 분자 생물학의 발전으로 인해서 생체 분자를 이용하여 계산을 수행하고자 하는 DNA computing 기법에 대한 연구가 활발해지기 시작했다. 1994년 Adleman이 NP-complete 문제인[1] 해밀토니안 경로 문제(Hamiltonian Path Problems: HPP)를 생물학적 과정만으로 해결함으로써 새롭게 DNA computing 기법을 이용한 최적해 문제에 대한 연구가 활발해 졌다. 지금까지의 인공지능에서는 신경망이나 진화 연산처럼 대부분 생물학적 개념만을 이용해서 계산 모델을 만들어 이를 적용하여 왔다. 그러나 DNA computing 기법은 실제 생체 분자인 DNA를 계산의 도구 및 정보 저장 도구로 사용하는 새로운 방법으로 진화 연산과 결합하여 인공지능의 새로운 한 분야가 되었다. 현재까지도 DNA가 가지고 있는 막대한 병렬성을 이용하여 최적화 문제들을 해결하고자 하는 연구들이 많이 진행 되고 있다. DNA computing 에서는 A(Adenine), C(Cytosine), G(Guanine), T(Thymine) 4가지 염기로 정보를 표현한다. 즉 0과 1의 2진수를 사용하는 유전알고리즘(Genetic Algorithms: GA)과는 달리 4가지 염기를 사용하는 4진수를 사용한다. 또한 막대한 병렬성을 이용하여 주어진 탐색 공간을 효율적으로 탐색할 수 있다. GA는 Holland의 저서에서[2] 처음으로 소개되었으며, 스트링의 개체 집단 위에서 모의 진화를 일으켜 효율적으로 최적 해를 탐색하는 알고리즘이다. 두 부모의 유전자로부터 그들 자손의 유전자를 형성하는 유성생식과 자연환경에서 일어나는 진화원리를 흉내내고있다.

본 논문에서는 인위적인 DNA의 개발 메커니즘에 근거를 둔 DNA coding기법과 GA를 사용하여 전역 최적해를 탐색하는 모의실험을 수행하였다. 연산자 및 각 연산자의 파라미터 등과 같은 조건들은 모두 동일하게 적용하여 각각의 성능을 조사하였다.

접수일자 : 2001년 9월 15일
완료일자 : 2001년 12월 1일

2. DNA Coding 방법 및 알고리즘

2.1 생물학적 DNA

모든 생명체는 각각 고유의 DNA를 가지고 있다. 그림 1은 DNA는 개체의 특성을 발현시키는 유전코드를 보여주고 있다. DNA는 A, T, G, C 4종류의 염기 배열로 이루어져 있으며, 이 유전코드는 A≡T, G≡C의 수소 결합으로 된 2중 나선구조를 가지고 있다. 그리고 2중 나선은 서로 3'에서 5'로 5'에서 3'으로의 서로 반대 방향으로 상보 결합을 이루고 있다.

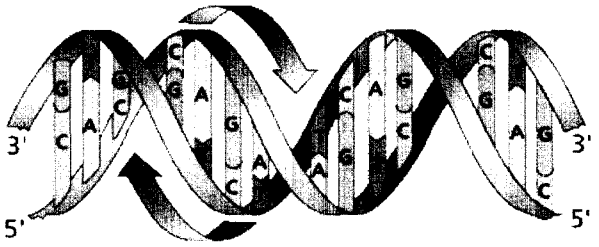


그림 1. 생물학적 DNA 구조
Fig 1. Structure of Biological DNA

A, T, G, C 네 종류의 염기배열 중 세 개의 배열이 한 의미단위를 이루어 해석된다. 이 의미단위를 생물학적인 용어로 코돈(codon)이라 하며, 이는 유전 정보의 최소단위가 된다. 총 64종류의 코돈은 20종류의 아미노산이 된다. 코돈의 64종류의 패턴에 대하여 생성되는 아미노산이 20종류인 이유는 다른 코돈이 같은 아미노산을 만들기도 하기 때문이다. DNA는 RNA로 전사되어 리보솜에서 단백질로 번역된다. 즉 아미노산을 암호화하는 DNA의 배열에 따라 아미노산의 합성순서를 결정하여 여러 종류의 단백질을 만들어낸다. RNA의 단백질로의 번역은 AUG에서 시작되어 UGA에서 번역이 끝난다. DNA에서는 U대신 T를 사용한다. 표1은 RNA 코돈과 생성하는 아미노산에 대한 것을 보여준다.

표 1. RNA(DNA) 코돈과 생성하는 아미노산
Table 1. RNA(DNA) Codon and Amino Acid

	U		C		A		G		
U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys	U
	UUC		UCC		UAC		UGC		C
	UUA	Leu	UCA		UAA	정지	UGA	정지	A
	UUG		UCG		UAG		UGG	Trp	G
C	CUU		CCU	Pro	CAU	His	CGU	Arg	U
	CUC		CCC		CAC		CGC		C
	CUA		CCA		CAA	Gln	CGA		A
	CUG		CCG		CAG		CGG		G
A	AUU	Ile	ACU	Thr	AAU	Asn	AGU	Ser	U
	AUC		ACC		AAC		AGC		C
	AUA		ACA		AAA	Lys	AGA	Arg	A
	AUG	Met	ACG		AAG		AGG		G
G	GUU	Val	GCU	Ala	GAU	Asp	GGU	Gly	U
	GUC		GCC		GAC		GGC		C
	GUA		GCA		GAA	Glu	GGA		A
	GUG		GCG		GAG		GGG		G

2.2 DNA Coding 기법과 GA 알고리즘

그림 2는 DNA coding기법과[3-4] GA의 최적해를 탐색하는 전체적인 알고리즘을 보여주고 있다. 전체적인 알고리즘을 살펴보면 다음과 같다.

1) 문제를 표현하는 초기 해 집단을 random하게 생성한다.
2) 해 집단의 적합도를 구한다.
3) 루울렛 휠 선택자를 구현하여 최종 해가 될 가능성이 없는 해들을 삭제하고 가능성이 높은 해들만 보존하여 해를 진화시킨다.
4) 교배와 돌연변이 연산자를 수행한다. 교배는 2점 교배를 하고 국소 해에 빠질 위험성을 벗어나기 위해 random하게 교배 점을 선택한다. 돌연변이는 모든 코드에 대하여 수행하며, 교배와 돌연변이는 모두 주어진 확률 값에 의해 행해진다.
5) 현재 세대수가 최대 세대수와 같으면 알고리즘 수행을 마친다.

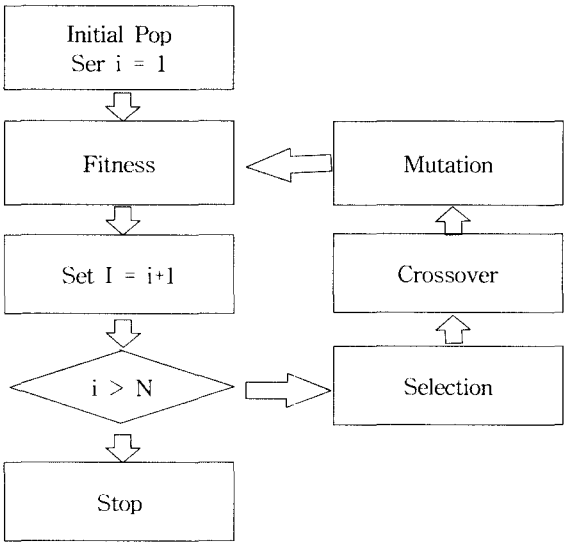


그림 2. DNA Coding 및 GA에서 사용된 알고리즘
Fig 2. Algorithm Used in DNA Coding and GA

2.3 DNA, GA의 Coding 방법

DNA는 A, T, G, C의 4종류의 염기 배열중 세 개의 배열이 한 의미단위가 되기 때문에 나타낼 수 있는 codon은 총 64종류가 있으며, 이는 다시 20종류의 아미노산이 된다. 아미노산들은 각자의 중요한 의미를 가지기 때문에 염기 배열을 유전정보, 또는 유전 암호라고 한다. 유전코드는 ATG에서 시작하고 종료 codon TAG에서 끝난다.

그림 3은 DNA 염색체의 예와 변환 메커니즘을 보여준다. 유전자는 시작 codon ATG에서 시작하고 종료 codon TGA에서 끝난다. 각각의 codon에 대응하는 아미노산들은 문제해결을 위한 자신의 역할을 갖는다. 또한 중복 유전자들은 중요한 의미를 갖는다. 그림 4는 한 염색체에서 유전자의 중복을 보여주고 있는데, 여기서는 하나의 DNA 내에 3개의 gene가 존재하며 gene5는 gene3, gene4와 중복되어 나타나 있음을 보여주고 있다. 이러한 중복 유전자에 의한 gene의 표현이 DNA coding에서의 장점중의 하나이며 GA에 비해서 다양한 표현 가능성을 보여준다.

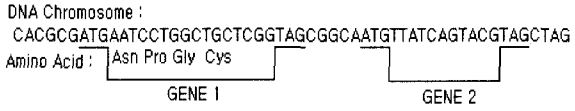


그림 3. 염색체의 변환 메커니즘의 예
Fig 3. Example of a Chromosome and Translation Mechanism

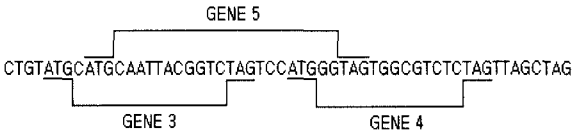
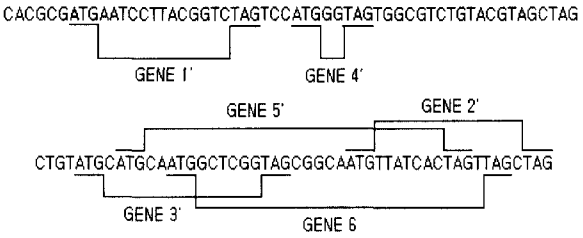
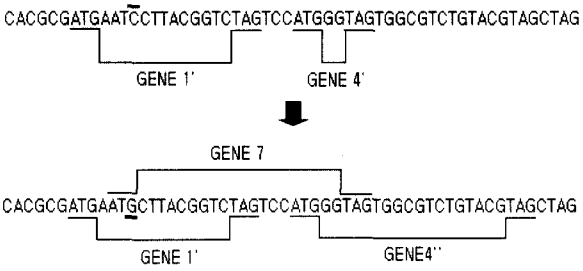


그림 4. 유전자의 중복의 예
Fig 4. Example of Overlapping of Genes

그림 5는 교배의 예를 보여준다. 교배는 주어진 교배확률에 의해서 발생하며 본 논문에서는 이점 교배 (two-point crossover)를 하여 두 부모 염색체 안에 분산되어 있는 어떤 유전정보를 결합하지 못하는 일점 교배(one-point crossover)의 단점을 보완하도록 했다. 그림 5(b)는 돌연변이의 예로, 돌연변이 확률에 따라 돌연변이 연산자에 의해서 C가 G로 바뀌었음을 보여준다. 또한 이와 같은 돌연변이 결과로 gene 7이 새로 생겼음을 알 수 있다. 돌연변이는 random하게 A, T, G, C중 하나로 바뀌게 하였다.



(a) 교배
(a) Crossover



(b)돌연변이
(b) Mutation

그림 5. 교배와 돌연변이의 예
Fig 5. Examples of Crossover and Mutation

3. 실험 및 결과

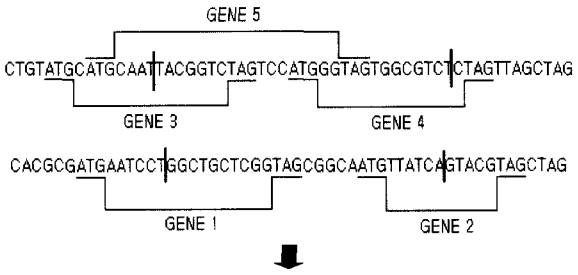
3.1 모의 실험

DNA coding 기법과 GA의 성능을 비교하기 위해서 다음과 같은 multi-modal 함수를 사용하였다.

$$f(x) = x + |\sin(32 \times x)| \qquad (1)$$
$$0 \le x \le \pi$$

그림 6은 모의실험이 사용된 식(1)에 의한 그래프이며 본 그래프에는 국부 최대점이 많이 있음을 보여주고 있다. 본 모의실험에서는 주어진 식(1)에서 x의 범위 내에서 $f(x)$가 최대가 되는 전역 최대점을 DNA coding 방법과 GA로 찾아서 각각의 성능을 조사하였다.

DNA coding에 있어서 코돈은 각자의 의미를 가지고 있다. 하나의 시작 코돈을 가지고 진화를 하는 것은 염색체를 비효율적으로 사용하는 것이 된다. 그래서 시작 codon을 ATG 대신 AT*(ATT, ATC, ATA, ATG)의 4종류로 지정하고 종료 코돈은 지정하지 않았으며 종료 코돈은 0에 대응되도록 했다.



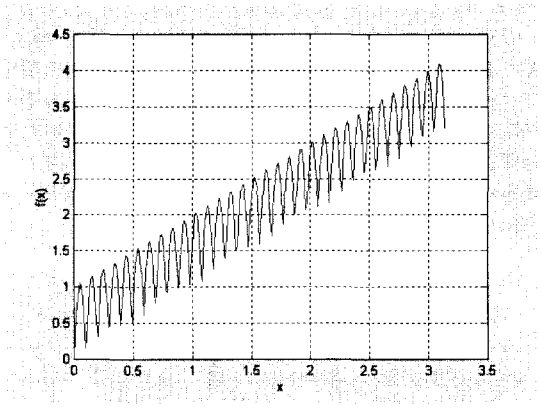


그림 6. 모의실험에 사용된 함수
Fig 6. Multi-Modal Function Used in Simulation

표2는 아미노산과 이에 해당하는 코드를 보여주고 있다. 첫 번째 아미노산은 소숫점 이상의 값을 코딩하고 두 번째 이후의 아미노산은 소숫점 이하의 값을 코딩하도록 하여, 이 코드들의 결합에 의해서 식(1)의 가능한 해들을 표현하도록 하였다.

표 2. 각 아미노산에 부여된 코드
Table 2. Code for Each Amino Acid

Phe	0	Pro	4	His	4	Glu	7
Leu	1	Thr	5	Gln	5	Cys	8
Ile	0	Ala	6	Asn	6	Trp	0
Met	2	Tyr	2	Lys	7	Arg	8
Ser	3	Val	2	Asp	7	Gly	9

모의실험에 사용된 각 파라미터들은 표 3과 같이 DNA coding 방법과 GA에서의 조건을 같게 하였다. 각각의 세대수는 50세대로 하였으며 집단의 크기는 60, crossover 확률은 0.8, mutation 확률은 0.1로 주었다.

표 3. 모의실험에 사용된 파라미터 값
Table 3. Parameters Used in Simulation

	DNA	GA
세대수	50	50
집단 크기	60	60
염색체 길이	300	20
Crossover 확율	0.8	0.8
Mutation 확율	0.1	0.1

3.2 결과 및 고찰

표 4는 DNA coding 방법과 GA를 이용했을 경우에 각각 50 세대 후에 찾은 최적해와 그때의 함수 값을 보여주고 있으며, 그림 7은 DNA coding방법을 사용했을 경우와 GA를 사용했을 경우에 있어서의 각 세대에서의 $f(x)$의 최대 값(Objbest)을 나타내고 있다. 또한 그림 8은 각 세대에서 해의 평균(Objave)을 나타내고 있다. 그림 7과 8에서 X 축은 세대수를 나타내며 Y축은 $f(x)$와 x의 평균을 나타낸다. 표 4에 의하면 DNA coding 방법으로 찾은 결과는 4.0929, GA를 이용하여 찾은 결과는 4.0909의 값을 최대 값으로 찾았으며, 이는 DNA coding 방법에 의한 해가 GA에 의한 해보다 약 0.05%의 작은 성능이 향상되었음을 보여주고 있다. 하지만 그림7과 그림8에서 보면 DNA coding 방법은 초기 세대에서부터 좋은 값을 구하는 반면, GA에서는 약 30세대 후에서야 DNA coding 방법에 의해 구한 최적해와 유사한 값을 구하였다. DNA coding 방법이 GA에 비해서 좋은 결과를 얻을 수 있는 이유는 DNA coding 방법이 다음과 같은 장점들을 가지고 있기 때문이다.

첫째, 0과 1의 2진수를 사용하는 GA에 비하여 DNA coding 방법은 A, T, G, C의 4가지 염기를 사용하여 코딩하기 때문에 해의 표현이 다양하다.
둘째, DNA coding 방법에서는 coding에 여분이 있으며 또한 중복되어 해를 표현할 수 있다.
셋째, 염색체의 길이가 가변적이다.

이와 같은 DNA coding의 특성은 GA가 갖는 전역탐색능력과 더불어 해의 표현방법이 다양하다는 장점을 더해주고 있다.

표 4. DNA coding 방법과 GA에 의한 최적값
Table 4. Optimum Value of $f(x)$ using DNA Coding Method and GA

	DNA	GA
최적해(x)	3.093413	3.095528
$f(x)$의 값	4.092991	4.090853

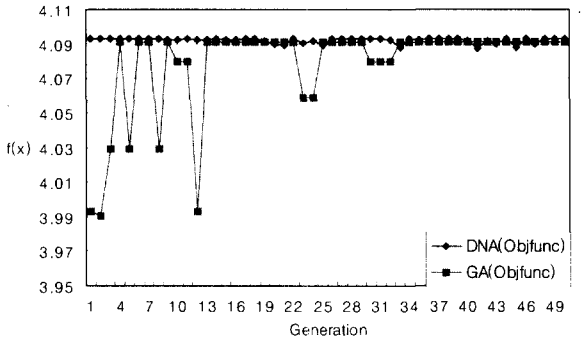


그림 7. 세대별 $f(x)$의 최대값
Fig 7. Maximum Value of $f(x)$ at Each Generation

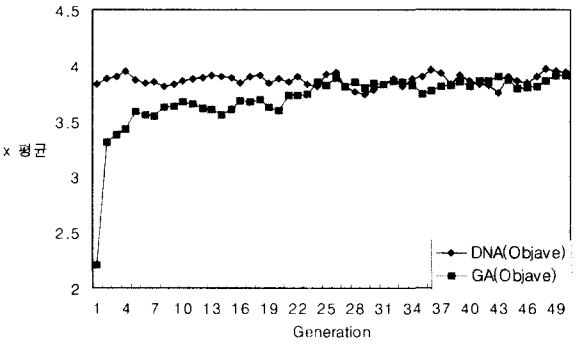


그림 8. 세대별 x값의 평균
Fig 8. Average Value of x at Each Generation

4. 결 론

본 연구에서는 최적해를 구하는 문제에 있어서 DNA coding 방법과 GA의 성능을 비교하여 보았다. 두 가지 방법 모두 최적해를 효과적으로 찾았으며, DNA coding 방법에 의한 결과가 약간의 성능향상이 있음을 알 수 있었다. 여분의 특성과 유전자 중복의 허용성 때문에 지식의 표현이 다양한 DNA coding 방법에 의한 방법에서는 주어진 multi-modal 함수의 최적값을 GA에 비하여 매우 적은 세대에 찾는 우수한 결과를 보여주었다.

DNA coding 방법은 DNA 분자의 막대한 병렬성과 여러 gene가 하나의 염색체내에 있을 수 있다는 장점 때문에 복잡한 문제에 적용하였을 경우 우수한 결과를 얻을 수 있을 것으로 예상된다.

향후 연구에서는 교배 위치에 따른 탐색 성능의 관계, 본 논문에서 사용한 연산자 외 화학적인 연산자의 적용, 염기의 종류의 연구와 패턴 인식에 있어서의 DNA coding 방법의 적용방법과 효율성에 대한 연구가 이루어져야 할 것이다.

참 고 문 헌

[1] Leonard M. Adleman, "Molecular Computation of Solutions To Combinatorial Problems", *Science*,

pp. 159-171, 1996
[2] J. H. Holland, Adaptation *in Natural and Artificial Systems*, The University of Michigan Press, 1975
[3] Tomohiro Yoshikawa, Takeshi Furuhashi and Yoshiki Uchikawa, "The Effects of Combination of DNA Coding Method with Pseudo-Bacterial GA," *Proc. IEEE Int. Conf. Evolutionary Computation*, Indianapolis, IN, USA, pp. 285-290, April, 1997
[4] Gheorghe Paun, Grzegorz Rozenberg, Arto Salomaa, *DNA Computing-New computing Paradigms, Springer*, Berlin, July 1998
[5] M. Amos, *DNA Computing*, Ph.D. thesis, The University of Warwick, UK, September 1997
[6] Brian Hayes, "The Invention of The Genetic Code," *American Scientist*, January-February, 1998
[7] R. Deaton et. al, "A DNA Based Implementation of an Evolutionary Search for Good Encodings for DNA Computation," *Proc. IEEE Int. Conf. Evolution Computation*, Indianapolis, IN, USA, pp. 267-271, April, 1997
[8] Piotr Wasiewicz, Tomasz Janczak, J. Mulaka, "The Inference via DNA Computing," *IEEE*, pp. 988-993, 1999
[9] Sungyong Yun et. al, "Acquisition of Fuzzy Rules Using DNA Coding Method," *한국퍼지 및 지능시스템학회 '98 춘계학술대회 학술발표 논문집*, vol. 8, no. 1, pp.16-19, 1998
[10] Dong-Wook Lee, Kwee-Bo Sim, "A Characteristics of Cellual Automata Neural Systems," *한국 퍼지 및 지능시스템 학회 추계학술대회 논문집*, pp. 267-272, 1998. 11월

저 자 소 개

백동화 (Paek, Dong-Hwa)
2001년 : 대구 가톨릭대학교 자동차 전자 공학과 졸업
2001～현재 : 명지대학교 전기 정보 제어공학부 석사 과정

관심분야 : 신경회로망, 유전알고리즘, DNA Computing, Pattern Recognition.

Phone : 016-786-1663
E-mail : fog0577@hanmail.net

강환일(Kang, Hwan Il)
1980년 : 서울대 전자공학과 졸업
1982년 : 한국과학기술원 전기 및 전자공학과 졸업(석사)
1992년 : 미국위스콘신 매디슨 대학 전기 및 전자공학과 졸업(박사)
1996년～현재 : 명지대 전기정보제어공학부 부교수

관심분야 : 퍼지 이론, 신경회로망, 유전알고리즘.

Phone : 031-330-6476
Fax : 031-321-0271
E-mail : hwan@mju.ac.kr

김갑일 (Kim, Kab Il)
1979년 : 서울대 전기과 졸업.
1981년 : 한국과학원 전기 및 전자공학과 졸업(석사)
1990년 : 클램슨대학교 전기 및 컴퓨터공학과 졸업(박사)
1981～1985년 : 육군사관학교 전자공학과 전임강사
1991～현재 : 명지대학교 전기정보제어공학부 교수.

관심분야 : 로봇공학, 자동화 시스템, 워터마킹, 산업통신 시스템, 웨이브렛 변환, 제어공학.

Phone : 031-330-6356
Fax : 031-321-0271
E-mail : kkl@mju.ac.kr

한승수 (Han, Seung-Soo)
1986년 : 연세대학교 전기공학과 졸업
1988년 : 연세대학교 전기공학과 졸업(석사)
1996년 : 조지아공대 전기 및 컴퓨터 공학과 졸업(박사)
2000년～현재 : 명지대학교 전기정보제어공학부 조교수

관심분야 : 신경회로망, 유전알고리즘, DNA Computing, Pattern Recognition, 정보보호.

Phone : 031-330-6345
Fax : 031-321-0271
E-mail : shan@mju.ac.kr