한국 퍼지 및 지능시스템학회 논문지 2000, Vol. 10, No. 4 (2000)315~323

시계열 예측을 위한 DNA코딩 기반의 신경망 진화

Evolutionary Neural Network based on DNA coding method for Time series prediction

이기열 · 이동욱 · 심귀보

Ki-Youl Lee, Dong-Wook Lee and Kwee-Bo Sim

중앙대학교 전자전기공학부

요 약

본 논문에서는 생명창발과 진화에 기반한 신경망 구성방법을 제안한다. 이 방법은 생물의 DNA 구조의 특성과 식물의 생장에 기반을 둔 방법이다. 본 논문에서 제안한 방법은 DNA 코딩 방법과 L-system의 생장 규칙을 이용하여 신경망을 구성하는 방법이다. L-system은 병렬적인 재조합 규칙을 이용하며, DNA 코딩 방법은 표현의 제약이 없는 표기법이다. 또한 진화 알고리듬은 다윈의 자연도태를 모방한 탐색법으로 다양한 해공간의 표현과 높은 효율로 탐색이 가능하다. 본 논문에서는 이러한 방법들을 이용해 신경망을 구성하고, 신경망의 Mackey-Glass, Sunspot, KOSPI 같은 시계열 예측문제에 적용하여 유효성을 입증하고자 한다.

ABSTRACT

In this paper, we propose a method of constructing neural networks using *bio-inspired emergent and* evolutionary concepts. This method is algorithm that is based on the characteristics of the biological DNA and growth of plants. Here is, we propose a constructing method to make a DNA coding method for production rule of L-system. L-system is based on so-called the parallel rewriting mechanism. The DNA coding method has no limitation in expressing the production rule of L-system. Evolutionary algorithms motivated by Darwinian natural selection are population based searching methods and the high performance of which is highly dependent on the representation of solution space. In order to verify the effectiveness of our scheme, we apply it to one step ahead prediction of Mackey-Glass time series, Sunspot data and KOSPI data.

1. 서 론

이 세계는 많은 시스템들의 집합으로 이루어져 있다. 시스템은 개개의 독립적인 시스템으로 존재 할 수도 있고, 여러 개의 독립 시스템들의 상호 작용으로서 하며 존재할 수도 있다. 시스템은 선형적인 출력을 내는 경우도 있으나 대부분은 비선형적인 출력을 낸다. 이러한 시스템의 출력을 예측하는 것은 시스템을 제어하는데 대단히 중요한 요소가 된다.

하나의 시스템을 출력 혹은 행동을 예측하는 방법은 크게 두 가지로 구분할 수가 있다. 첫 번째 방법은 모델에 기초한 방법이고 두 번째는 통계에 의한 방법이다.

모델에 기초한 방법은 시스템에 대한 충분한 정보가 있다는 가정하에 그 시스템에 대한 정확한(때로는 특정 환경에서 만이라도) 수학적으로 증명된 모델을 구성하고, 이 모델을 통해 시스템의 출력을 예측하는 것이다. 이 경우에는 모델이 충분히 정확하지 않다면 예측에 대한 결과는 신뢰할 수가 없게 된다. 즉, 가장 중요한 요소는 정확한 모델을 세우는 것인데, 이를 실제 세계에 적용하기에는 많은 제약이 있다. 현재 우리가 가진 지식과 기술로는 모델을 세우는데 충분한 정보를 획득하기가 어려울 뿐만이 아니라 정보가 있다하여도 이를 바탕으로 모델을 세우기도 대단히 어렵다. 실제로 대부분의 시스템들은 강한 비선형성을 가지며, 복잡하고 불규칙한 신호를 발생시키는 경향이 있다.

통계에 의한 방법은 시스템의 과거 출력 값을 분석하여 이를 미래에 대한 예측 자료로 사용하는 방법이다. 통계적인 방법으로는 수학적인 모델의 형태로 표현하는 것은 어렵다. 하지만 수학적인 모델의 구성이 어렵기 때문에 현대의 많은 예측 방법들이 통계적인 방법에 의해 개발되었다.

통계적인 방법은 특정한 수학적 모델이 없이 구성을 하기 때문에 일반적인 진화 연산기법이나 신경망

본 연구는 과학기술부의 뇌과학 프로젝트(Braintech21)의 지원으로 이루어진 결과임. (과제번호 : 98-J04-01-01-A-07)

을 이용한 방법이 많이 사용된다. 특히 Genetic Programming(이하 GP)에 의한 방법은 수학적인 표현이 가능하다는 점에서 많이 사용되며, 신경망에 의한 예측 방법은 강한 비선형성을 나타내기에 많이 사용하는 방법이다. 하지만 GP에 의한 방법은 코드가 지나치게 길어진다는 단점이 있고, 신경망은 학습이라는 장점이 있으나 구조의 변경이 불가능하다는 단점이 있다.

이러한 단점을 보완하고자 본 논문에서는 적은 코드로도 많은 정보의 표현이 가능한 DNA 코딩법과 신경망 구성을 위한 L-system, 그리고 신경망에 진화연산 기법을 도입하여, 시계열 예측문제에 적용하였다.

진화 알고리즘은 개체군중에서 개체의 적합도를 평가한 후 다음 세대에서 적합도가 높은 개체가 살아남는 자연도태의 방식을 택한 알고리즘이다. 또한 발생모델의 L-system은 일반적인 초기 규칙과 생성규칙으로 구성된다. 이들 규칙을 바탕으로 하나의 시스템을 구축하기 때문에 적은 규칙만으로도 복잡한 개체를 생성할 수 있다.

본 논문에서는 원하는 목적의 신경망을 얻기 위하여 DNA 코딩[4]을 이용하여, 규칙을 생성시킨 후 그 규칙을 통해 신경망을 구성한다. 또한 진화알고리즘을 이용하여 점점 더 우수한 개체를 선택함으로서 최종적으로 원하는 목적의 신경망을 구현한다.

DNA 코딩은 생물학적인 DNA 구조를 모방한 것으로서 DNA가 생물의 유전정보를 통해 자신을 발생시키고, 또한 다음 세대에 유전 정보를 전달하는 과정을 모방한 방법이다. DNA코딩은 동적인 구조를 통한 중복 해석과 여분이 있다는 장점 때문에 적은 양의 DNA만으로도 많은 정보를 가지고 있을 수 있다. 이러한 특징을 이용하여 하나의 해석 단위를 하나의 신경망의 노드와 그의 부수적인 요소(Bias, Weight, 입출력 범위)를 결정하게 한다. 그리고 중복 해석을 통해 나온 규칙(L-system의 생성규칙 이용하여 구성)을 이용하여 신경망의 구조를 결정한다.

이렇게 자동 생성된 신경망을 Mackey-Glass 시계열 예측 문제와 Sunspot 데이터, KOSPI 데이터의 예측에 적용시켜, 우수한 개체를 선택한 후 이들 개체의 유전자를 GA 연산자(돌연변이와 교배)를 통하여 다양화된 유전자를 다음 세대로 전달하고, 더 좋은 개체를 얻도록 하여 최종적으로 시계열 예측문제를 풀 수 있는 신경망을 자동적으로 생성하도록 하는 것이다.

2. DNA 코딩

2.1 생물학적 DNA

모든 생물체는 각자 고유의 DNA를 가지고 있다. DNA는 개체의 특성을 발현시키는 유전코드로서, A(아데닌) T(티민, RNA에서는 U : 우라실) G(구아닌) C(시토신)의 4개의 염기배열로 이루어져 있다. 또한 염기 3개의 배열이 한 의미단위를 이루어 해석된다. 이 의미단위를 생물학적인 용어로 코돈(codon)이라 한다. 코돈의 가짓수는 4×4×4 = 64개이며 이것이 코드화 하는 아미노산은 20가지이다. 코돈의 64가지 패턴에 대하여 생성하는 아미노산이 20가지인 이유는 다른 코돈이 같은 아미노산을 만들기도 하기 때문이다. 이것은 표 1에 나타나 있다[5].

DNA는 RNA로 전사되어 리보솜에서 단백질로 번역된다. 즉 아미노산을 암호화하는 DNA의 배열에 따라 아미노산의 합성순서를 결정하여 여러 종류의 단백질을 만들어낸다. RNA의 단백질로의 번역은 AUG에서 시작해서 UGA(UAA, UAG)에서 번역이 끝난다.

표 1. RNA(DNA) 코돈과 생성하는 아미노산(DNA에서는 U대신 T를 사용한다.)

	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

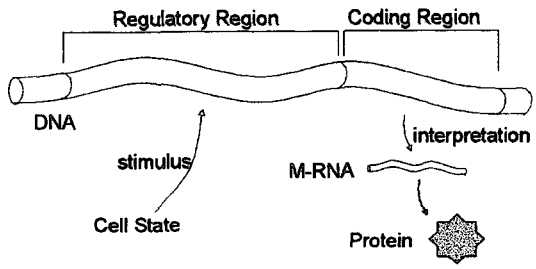


그림 1. 유전자 구조의 일반적인 구조

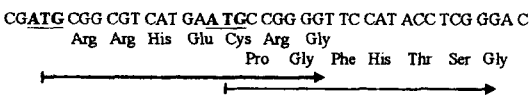


그림 2. DNA 염색체의 번역

생물학적 유전자의 기본적인 구조는 그림 1과 같다. 즉, 단백질을 직접 코드화 하는 부위(coding region)와 그 코드 부위의 발현을 조절하는 조절부위(regulatory region)로 구성되어 있다. 코드화 부위는 조절부위의 명령에 의해 세포내의 조건이 조절부위를 자극할 때 단백질로 번역된다. 이것은 발생모델의 규칙의 표현방식과 유사하다. 즉, 조절부위는 규칙의 전건부 또는 세포의 현재 주변 상태, 코드화 부위는 후건부 또는 세포의 다음 상태에 대응된다.

2.2 DNA 코딩방법의 특징

DNA의 동작을 모방한 코딩방법에 대한 연구는 Yoshikawa[4] 등에 의하여 기본적인 방법이 연구되었다. 염색체는 기본적으로 4가지 염기의 배열로 이루어져 있고 아미노산을 번역하는 것과 마찬가지로 코돈 단위로 번역한다. 또한 다음 그림 2와 같이 유전자의 번역 시작점이 일정하지 않기 때문에 중복 번역을 허용하며 때에 따라서는 번역되지 않는 부분도 존재한다. 표 1에서 알 수 있듯이 하나의 아미노산을 생성하는 코돈이 여러 개이기 때문에 염색체의 중복을 효율적으로 이용할 수 있다. 교차점도 임의로 주어지기 때문에 염색체의 길이도 가변적이다. Yoshikawa의 방법은 번역 테이블을 만듦으로서 규칙의 표현에 적합하게 구성되어 있다. 이상의 특징을 정의하면 아래와 같은 4가지로 정리할 수 있다.

◆ DNA 코딩방법의 특징

(1) 지식의 표현이 쉽다.
(2) 코딩에 여분과 중복이 있다.
(3) 염색체의 길이가 가변적이다.
(4) 교차점에 제약이 없다.

Wu[6] 등은 GA에서의 가변위치 표현법(floating representation)에 대한 스키마 분석을 통하여 유효성을 증명하였다. 가변위치 표현법이란 DNA 코딩에서와 같이 시작 기호(start tag)를 가지고 있어서 시작 기호가 발견되는 곳에서부터 유전자의 번역을 시작하는 방법이다. Wu는 비록 비트 스트링을 가지고 실험을 하였으나 이 방법도 DNA 표현 방법의 (2), (3), (4)의 특징을 모두 가지고 있다. 유용한 스키마를 찾는 능력에 대한 일반 GA방법(고정위치 표현법)과의 비교에서 그는 다음과 같은 결론을 얻어냈다. (a) 염색체의 길이가 짧을 경우에는 일반적인 고정위치 표현방법(fixed representation)이 우수했으나 염색체의 길이가 길수록 가변위치 표현법의 성능이 더 좋아진다. (b) 가변위치 표현법에서 개체군의 다양성이 더욱 높다. 따라서 매우 복잡한 문제에 대하여 적은 수의 개체군을 가지고 개체의 다양성을 최대로 유지하며 탐색을 수행한다.

그림 2에서 보듯이 위와 같은 DNA코드가 있다면, 그 안에 있는 시작코돈(ATG)부터 해석을 시작하여 각각의 아미노산을 해석하여 그에 따른 규칙을 생성한다. 이와 같은 해석 방법을 통하여 짧은 DNA코드에서도 많은 정보를 얻을 수 있다.

3. L-System

L-system[1-3]은 병렬적인 문자열의 재조합 속성을 갖는 일종의 문법으로서 1968년 Aristid Lendenmayer에 의해 제안되었다. 이것은 다세포 생물의 성장 과정의 모델링이 가능하기 때문에, 이후 컴퓨터 그래픽 등에서 식물을 모델링하는데 많이 이용되고 있다.

3.1 Simple L-system

L-system은 초기문자열(axiom)로부터 생성규칙(production rule)의 반복적인 적용에 의하여 생성된 최종 문자열은 심벌(symbol)의 문맥에 따라 여러 가지 방식으로 해석되며, 일반적으로 선을 그리는 방식을 택하여 나무 모양을 만들어낸다. 간단한 L-system의 구성요소들은 다음과 같이 정의한다.

· 문자(Alphabet) Σ : 심벌들의 유한집합으로, 주로 a, b, c 같은 문자들이 쓰지만 다른 문자라도 상관이 없다.

ex) Σ={a, b, c}

· 초기문자열(Axiom) α : 집합 V에서 정의된 심벌들의 연속된 문자열의 집합을 Σ^*라고 하면 초기문자열은 집합 Σ^*의 ϕ가 아닌 한 원소이며, 초기문자열부터 생성규칙에 의해 생장한다.

ex) Axiom = b

· 생성규칙(Producton Rule) Π : 하나의 심벌 $a(a \in \Sigma)$을 하나의 문자열 $w(w \in \Sigma^*)$로 대응시키는 것. 만약 특정 심벌에 대하여 어떤 생성규칙도 주어지지 않으면 자기 자신으로 대응시키는 것을 기본으로 한다.

ex) $a \rightarrow ab$
$b \rightarrow a$

언어(Language)로서 L-system의 문법 G는 식 (1)과 같이 표현한다.

$G = \{\Sigma, \Pi, , \alpha\}$ (1)

단, Σ은 문자의 집합, Π는 생성규칙의 집합($\Pi = \{\pi \mid \pi : \Sigma \rightarrow \Sigma^*\}$), α는 초기 문자열이다.

Example : $G = \{\Sigma, P, a\}$

$\Sigma = \{A, B, C\}$

$\Pi = \{A \rightarrow BA, B \rightarrow CB, C \rightarrow AC\}$

$\alpha = ABC$이면 최종적으로 생성되는 언어는

$L = \{ABC,$

$BACBAC,$

$CBBAACCBBAAC,$

$\cdot \cdot \cdot\}$

4. 신경망의 DNA 코딩방법

4.1 신경망 구성을 위한 DNA 코딩방법

본 절에서는 신경망을 진화시키기 위한 DNA코딩 방법을 설명한다. 신경망을 구성하기 위해 DNA 코드에서 규칙(L-system의 규칙)을 생성한 후, 그 규칙을 바탕으로 신경망을 구성한다.

우선 임의의 DNA 코드를 발생시킨다. 이 코드를 위 표에 의하여 해석을 하여 여러 개의 규칙을 만든다. 시작코돈(ATG)이 나오면 해석을 시작한다. 처음에 나오는 첫 코돈은 신경망의 문자로 해석을 한다. 두 번째 코돈은 신경망의 연결 범위를 결정한다. 예를 들어 숫자가 3이라면 최종적으로 만들어진 문자열에서, 그 노드로 3번째 노드까지 모두 연결이 된다. 단, 쉼표(,)가 나오면 그 옆의 노드는 연결하지 않는다. 세 번째 코돈은 신경망의 노드에 포함되는 Bias로 해석을 한다.

값의 계산은 Base(B) A=0, G=1, T=2, C=3으로 하여 식 (3)과 같이 계산을 한다. 이 후에 나타나는 코돈은 두 번째 코돈에서 정해진 숫자만큼 Weight 값으로 해석을 한다. 계산은 Bias와 같은 방법으로 한다. 여기까지 해석을 하면 하나의 신경망노드를 구성한 것이다. 같은 방법으로 정지 코돈이 나올 때까지 해석을 계속한다.

$$\text{Weight} = \frac{(B\times 4^2 + B\times 4^1 + B\times 4^0) - 32}{10} \quad (3)$$

$(-3.2 \le W \le 3.2 \quad 0.1$ 간격)

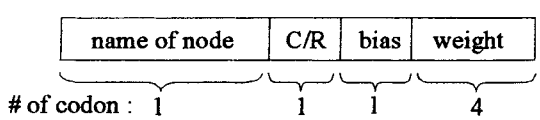


그림 3. 노드의 구성

표 2. DNA 코드표

Amino Acid	# of Amino Acid	Node's Name	Connecting Range
Leu	6	A	1,1
Arg	6	B	2,2
Ser	6	C	3,3
Thr	4	D	1,2
Ala	4	A	1,3
Gly	4	B	1,4
Val	4	C	2,3
Pro	4	D	2,4
Stop	3		
Ile	3	A	3,4
Tyr	2	B	4,4
Gln	2	C	1,1
Phe	2	D	2,2
Asp	2	,	3,3
Cys	2	,	1,2
Asn	2	,	1,3
Glu	2	,	1,4
His	2	,	2,3
Lys	2	,	2,4
Trp	1	C	3,4
Met	1	D	4,4

※Connecting Range(x, y) : 문자열(노드의 배열)에서, 현재 노드에서 연결할 수 있는 범위를 x번째 노드부터 y 번째 노드까지 정함. 단 쉼표(,)가 나올 경우 쉼표 뒤에 있는 노드는 연결하지 않으나 연결 범위에는 들어감.

이렇게 DNA 코드를 해석한 모드의 배열에서 첫 번째 코돈의 문자만을 뽑아 문자열을 만든다. 맨 처음에 나오는 문자를 L-system 규칙의 전건부로, 그 이후에 나오는 문자열은 후건부로 해석을 하여 L-system의 규칙을 생성한다. 이렇게 해석된 규칙은 문맥자유 L-system의 규칙과 같은 특성을 지닌다. 시작코돈이 나오는 개수 만큼 규칙이 나오기 때문에 여러 개의 규칙이 생성된다. 만약 여러개의 규칙 중에 같은 전건부를 갖는 규칙이 여러 개 나올 경우는 먼저 나온 규칙을 적용한다. 그 규칙을 가지고, 정해진 초기 문자열에 따라 규칙을 적용해 그 다음 단계의 문자열을 만들고, 맞는 규칙이 없을 경우에는 그 문자를 그대로 유지한다. 정해진 수만큼 반복하여 문자열을 치환한 후 생긴 최종 문자열을 가지고, 신경망을 구성한다.

이렇게 생성된 신경망은 반드시 한 개이상의 입력노드와 한 개이상의 출력노드를 갖는 무정형의 신경망이 된다. 문제에 따라 적당한 수의 입력노드와 출력노드를 갖는 신경망을 선택하여, 주어진 입력을 넣고, 그 출력을 검사하여 원하는 결과와 비교한 후, 유전자

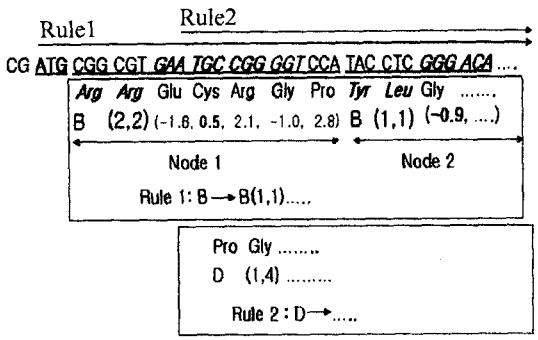


그림 4. DNA 코드의 해석

알고리즘에 의해 신경망을 진화시킨다. 신경망의 적합도는 식 (4), (5), (6)에 의해 구한다.

$$\text{Fitness} = \frac{1}{1+e^{-\lambda \cdot \delta^2(\tilde{x})}} \tag{4}$$

$$\delta^2(\tilde{x}) = \frac{1}{N}\times\sum_{t}^{N}|x(t+1)-\tilde{x}(x(y))|^2/var \tag{5}$$

$$var = \frac{1}{N}\times\sum_{t}^{N}\left|x(t)-\frac{1}{N}\times\sum_{t}^{N}x(t)\right|^2 \tag{6}$$

데이터들의 진동폭을 적합도에 적용함으로서 단순한 오차의 합으로 구하는 것과 예측 어려움의 정도를 적합도에 반영할 수 있다.

이 집단을 진화시키기 위하여 교배와 돌연변이, 진화전략(Evolution Strategy)의 (μ + λ)선택 방법을 사용하였다. 생성된 신경망의 DNA를 돌연변이와 교배를 통하여 3배수의 자손을 생성한 뒤, 원래 부모세대의 신경망과 같이 평가하여 순위선택으로 우수한 개체를 뽑아 다음세대의 부모 개체로 삼는다. 이 과정을 반복하여 점점 좋은 개체를 얻는다.

이 방법의 유효성을 검증하기 위해 Mackey-Glass 시계열 예측문제와 Sunspot 예측문제, 그리고 KOSPI의 데이터를 예측할 수 있는 신경망을 구성하였다.

4.2 신경망 생성의 예

XOR문제를 풀기 위한 신경망 생성을 예로 적용해 보도록 한다.

DNA에서 다음과 같은 5개의 규칙이 생성되었다고 가정을 해보자. 그러나 A를 전건부로 하는 규칙이 두 개가 나왔으므로 먼저 나온 하나만을 이용하여 스트링을 구성한다.

rule 1 : $A \rightarrow B(1,1)C(2,4)$
rule 2 : $B \rightarrow B(2,3)$

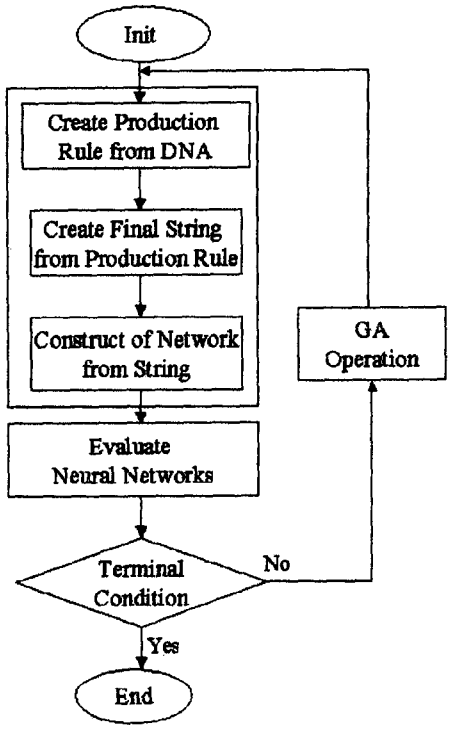


그림 5. 신경망의 진화

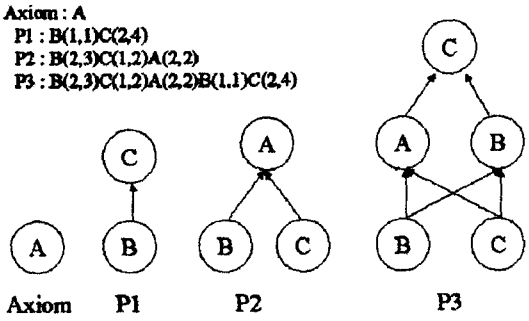


그림 6. 신경망의 구성과정

rule 3 : $A \rightarrow A(2,3)B(3)$
rule 4 : $D \rightarrow A(1,1)$
rule 5 : $C \rightarrow C(1,2)A(2,2)$

위의 규칙에 따라 3번 생장을 시키면 다음과 같이 변화하며

초기문자열 : A
1회 : $B(1,1)C(2,4)$
2회 : $B(2,3)C(1,2)A(2,2)$
3회 : $B(2,3)C(1,2)A(2,2)B(1,1)C(2,4)$

연결되지 않는 부분을 제거하여 새로이 문자열을 만

들면 최종적으로 $N(2,3)N(3,4)N(2,2)N(1,1)N(0,0)$의 문자열을 얻을 수 있다. 이러한 과정을 거쳐 XOR문제를 풀 수 있는 구조를 가진 신경망을 구성할 수 있다.

5. 시계열 예측을 위한 DNA 코딩 방법

5.1 시계열 예측

시계열 예측이란 과거에 얻어진 데이터를 가지고, 미래의 값을 예측하는 것이다. 시계열 예측방법으로는 신경망을 비롯해 Genetic Programming 등 여러 가지 방법이 있다.

주어진 과거 데이터를 x, 예측기를 통해 얻어진 결과를 $\tilde{x}$, 과거 데이터의 집합을 벡터 x라고 하면,

$$x(t) = (x(t),\ x(t-1),\ \cdots,\ x(t-\phi)) \tag{7}$$

로 표현할 수 있다. ϕ개 만큼의 과거 데이터를 이용하여, $x(t+1)$를 예측한 값, $\tilde{x}(x(t))$를 얻을 수 있다. 즉, 미래의 값, $x(t+\tau)$는 과거 값을 이용해 예측한 값 $\tilde{x}(x(t))$로 얻을 수 있다. 여기서 $\tau=1$이면 shot-term prediction이라하고, τ가 2이상이면 long-term prediction이라고 한다.

5.2 성능 비교

하나의 문제를 풀기 위하여 생성되는 신경망의 수는 대단히 많다. 이렇게 발생된 신경망을 하나의 예측기로서 동작을 한다. 이런 많은 예측기의 성능을 비교하기 위한 방법 중 하나로 Casdagli가 제안한 방법이 있다.

$$\delta^2(\tilde{x}) = \frac{1}{N}\times\sum_{t}^{N}|x(t+1)-\tilde{x}(x(t))|^2/var \tag{8}$$

$$var = \frac{1}{N}\times\sum_{t}^{N}\left|x(t)-\frac{1}{N}\times\sum_{t}^{N}x(t)\right|^2 \tag{9}$$

으로 나타낼 수 있다. 이 값은 RMSE(Root Mean Squared Error) 값으로 사용이 가능하다.

5.3 Mackey-Glass 시계열 예측문제

Mackey-Glass 함수는 카오스시스템의 대표적인 예로 식 (10)와 같이 표현된다.

$$\frac{dx(t)}{dt} = \frac{bx(t-\tau)}{1+x(t-\tau)^c} - ax(t) \tag{10}$$

Mackey-Glass 시계열 예측 문제의 경우는 DNA 길이 500으로 무작위로 발생된 초기개체군을 시작으로 진화를 시작하였다. 교배율은 0.9, 돌연변이율은 0.3으로 설정하였다. 교배방법은 일점(one point)교배 방식을 사용하였고, 2개 부모개체의 DNA에서 같은 위치에서 교차가 이루어지게 하여 DNA의 길이는 유지하도록 하였다. 그리고 개체의 선택방법은 Ranking Selection과 ES의 $(\lambda+\mu)$ 선택법을 혼합하여 선택하였다.

신경망의 입력은 과거 데이터 값으로 $x(t)$의 값을 구하는데 $x(t-1)\sim x(t-19)$의 값 중에서 구성된 신경망의 입력노드 수만큼 사용하였으며, 신경망 노드에서의 출력함수는 식 (11)을 사용하였다.

$$f(i) = \left(\frac{2}{1+e^{\eta\cdot i}}-1\right)\times 2 \tag{11}$$

입력노드는 5개에서 19까지 가질 수 있으며, 출력노드는 하나만 갖는 신경망을 평가 대상으로 하여 250개의 데이터를 넣고, 출력 값을 계산하여 오차가 적은 신경망을 선택하여 다음 세대의 부모 개체로 삼는다.

적합도의 변화를 보면, 적합도가 일정하게 유지되는 부분이 나타난다. 다양한 원인에 의해 이런 현상이 일어날 수 있지만, 그 중 하나의 원인으로 DNA 코딩의 특징 중 하나인 신경망으로 발현되지 않고 잠재된 부분의 변화가 일어난다. 이러한 변화가 누적되어 가

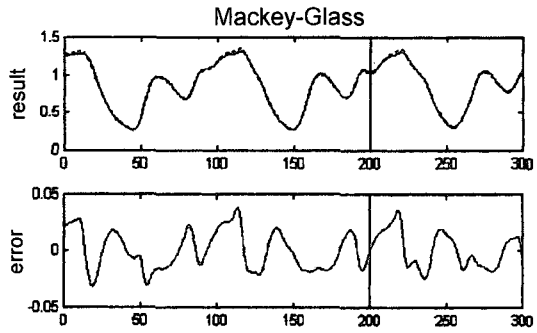


그림 7. Mackey-Glass 시계열 예측 결과
(··· ideal - predicted)

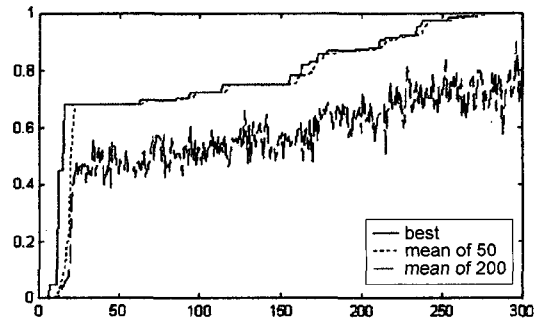


그림 8. Mackey-Glass 문제의 적합도 변화

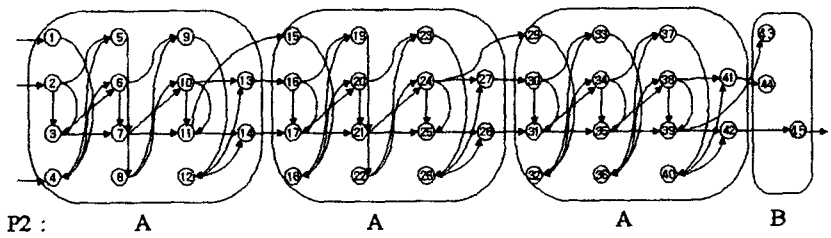


그림 9. 신경망의 예

며, 이러한 부분이 발현되면, 한 단계 높은 상태로의 진화가 급속도로 이루어진다.

그림 9는 Mackey-Glass 시계열 예측을 위해 진화한 L-시스템 신경회로망 중의 하나를 나타낸 그림이다. 이때 생성된 규칙은 다음과 같으며 3회 생장을 통하여 노드가 총 45개로 이루어진 신경회로망이 생성되었다.

Rule

R1 : A → AAAB

R2 : B → BC

R3 : D → BA

위의 규칙을 통하여 초기 문자열 A로부터 3회 생장시키면 최종적으로 P4의 문자열이 얻어진다.

String

P1 : A

P2 : AAAB

P3 : AAABAAABAAABBC

P4 :

AAABAAABAAABBCAAABAAABAAABBCA
AABAAABAAABBCBCCAAABAAABAAABB
CAAABAAABAAABBCAAABAAABAAABBC
BCCAAABAAABAAABBCAAABAAABAAAB
BCAAABAAABAAABBCBCC

5.4 Sunspot 시계열 예측문제

Sunspot 데이터는 1년 동안 태양의 흑점 수를 기록한 것으로, 일정한 규칙이나 흐름이 없이 관측된 과거 데이터만이 존재를 한다. 그러므로 예측방법도 과거의 데이터를 이용할 수밖에 없다. 본 논문에서는 Sunspot 예측문제를 제안한 신경망에 적용시켜 보았다. 200개의 학습데이터와 50개의 테스트 데이터를 사용하여 예측한 결과이다. 신경망의 발생을 위한 요소들은 Mackey-Glass문제와 동일하게 구성을 하였다.

5.5 KOSPI 예측문제

세 번째 적용문제는 국내 주식시세 예측문제이다.

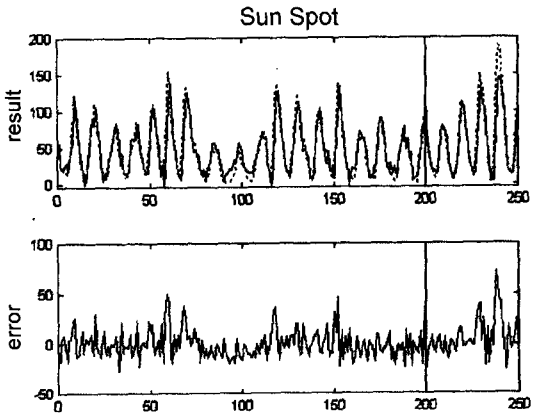


그림 10. Sunspot 시계열 예측 결과
(··· ideal - predicted)

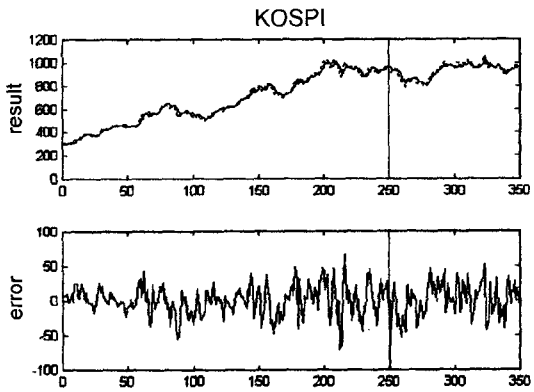


그림 11. KOSPI 데이터의 시계열 예측 결과
(···ideal - predicted)

예측을 위한 데이터는 1998년 2월부터 2000년 2월까지 얻은 350개 KOSPI(Korea Stock Price Index) 종합주가를 이용하였다. 250개를 진화를 위한 학습데이터로 나머지 100개를 테스트 데이터로 사용하였다.

5.6 비교

제안한 방식이 다른 방식을 이용한 예측결과와 어느정도의 성능을 나타내기 위하여 NMSE를 이용하여

표 3. NMSE를 이용한 시계열 예측의 성능 비교

예측 방법	Mackey-Glass	Sunspot
L-시스템 신경회로망 [본 논문]	2.8E-3(2.7E-3)	1.65E-2(1.58E-2)
FEP [Rao & Chellapilla 99]	2.63E-4	1.13E-2(1.89E-2)
CEV (standard) [Lapedes & Farber 87]	1.21E-2(1.29E-2)	
CEV (coevB) [Mayer & Schwaiger 99]	3.57E-2(3.62E-2)	

표 4. 시뮬레이션 파라메터

	MG	Sunspot	KOSPI
Population Size	50(50+150)		
Initial String Length	300		
Crossover Method	One point		
Crosover Prob.	0.9		
Mutation Prob	0.3		
Selection	Ranking & $(\mu+\lambda)$ selection		
Generation	300	500	500
# of input node	5~19		
# of output node	1		
Learning Data	250	200	250
Test Data	50	50	100
Range of output	-2~2		
Output Scale	×1	×100	×1000

성능을 비교해 본다. KOSPI 데이터는 성능을 비교할 수 있는 논문이 없기 때문에 Mackey-Glass 문제와 Sunspot 문제에 대하여 다른 방식의 시계열 예측 결과와 성능을 비교해 보았다[8,9].

본 논문에서 제안한 방법을 이용한 신경회로망을 이용하여 시계열 예측을 수행한 결과 FEP의 방법보다는 약간 성능이 떨어졌으나 나머지 두 방법에 비하여는 성능이 우수함을 알 수 있었다. FEP방법은 정밀한 실수치 탐색을 하는 방법으로 결과의 미세 탐색에 매우 유리한 방법이다. 반면 제안한 신경회로망의 구조는 정밀도 0.1의 연결강도와 바이어스를 가진 신경회로망을 이용해 예측을 하였기 때문에 FEP의 결과보다는 약간 성능이 떨어지는 것으로 생각된다. 따라서 신경회로망에 학습을 부가하여 미세한 실수치 연결강도를 구한다면 보다 성능을 향상시킬 수 있을 것이다. 그러나 이러한 점을 고려하면 전체적으로 비교적 우수한 성능의 결과를 얻음으로써 제안한 방법의 유효성을 알 수 있다.

6. 결 론

이 논문에서는 발생/발달 모델의 하나인 L-system과 DNA 코딩 방법을 이용하여 신경망을 진화시키는 방법을 제안하였다. DNA 코딩을 이용함으로서 L-system의 생성규칙을 짧은 DNA로 표현할 수 있다. 이렇게 생성된 규칙을 바탕으로 문자열을 구성한 후 이를 신경망으로 구성하였다. 이렇게 구성된 신경망을 Mackey-Glass 시계열 예측문제와 Sunspot 예측 문제에 적용하여 보았다.

이렇게 규칙을 통해 신경망을 구성함으로서, 작은 DNA의 변화로 신경망 전체에 큰 변화를 유도 할 수도 있다.

L-시스템 신경회로망은 L-시스템의 규칙을 이용해 신경회로망의 구조를 생성시키기 때문에 구조적으로 자기 유사성을 갖는 모듈형태가 얻어진다. 이러한 점 때문에 연결강도와 함께 신경회로망의 구조 자체가 신경회로망의 성능에 영향을 미치게 된다. 실험 결과에서 나타난 바와 같이 연결강도의 정밀도가 떨어짐에도 불구하고 우수한 결과를 얻어내었다. 또한 DNA 코딩은 L-시스템의 규칙은 표현에 거의 제약이 없는 형태이기 때문에 주어진 문제에 적합한 신경회로망을 찾는데 매우 적합함을 알 수 있다.

참고문헌

[1] P. Prusinkiewicz, M. Hammel, J. Wolters, R. Mech, "Visual Models of Plant Development," *Hand Book of Formal Languages,* Springer-Verlag, 1996.

[2] Aristid Lindenmayer, Prezemyslaw Prusinkiewicz, "Developmental Models of Multicellular Organisms : A Computer Graphics Perspective," *Artificial Life VI*, pp. 221-249, 1987.

[3] Aristid Lindenmayer, "Mathematical Models for Cellular Interaction in Development, Part I, II," *Journal of Theoretical Biology*, Vol. 18, pp. 280-315, 1968.

[4] T. Yoshikawa, T. Furuhashi, Y. Uchikawa, "A Combination of DNA Coding Method with Pseudo-Bacterial GA for Acquisition of Fuzzy Control Rules," *Proc. of 1st Online Workshop on Soft Computing, Aug.* pp. 19-30, 1996.

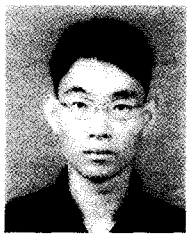
[5] R. A. Wallace, G. P. Sanders, R. J. Ferl, *BIOLOGY : The Science of Life 3rd eds.*, Harper Collins Publishers Inc., 1991.

[6] A. S. Wu, R. K. Lindsay, "A Comparison of the Fixed and Floating Building Block Representation in Genetic Algorithm," *Evolutionary Computation,* Vol. 4, No. 2, pp. 169-193, 1996.

[7] Casdagli, Martin, "Nonlinear Prediction of Chaotic Time Series", Phisica D, Vol. 35, pp. 335-356. 1989.

[8] Rao Sathyanarayan, S. etc Evolving Nonlinear Time-

Series Models Using Evolutionary Programming," CEC99 Vol. 1. pp. 236-243.
[9] Helmut A, Nayer. Roland Schwaiger "Evolutionary and Coevolutionary Approachs to Time Series Prediction Using Generalized Multi-Layer Predictions," CEC99 Vol. 1, pp. 275-280.



이 기 열 (Ki-Youl Lee)

1999년 : 중앙대학교 제어계측공학과 학사
1999년~현재 : 중앙대학교 제어계측학과 석사과정
관심분야 : 인공생명, 가상현실, 진화연산 등

이 동 욱 (Dong-Wook Lee)

1996년 : 중앙대학교 제어계측공학과 학사
1998년 : 중앙대학교 제어계측학과 석사
2000년 : 중앙대학교 제어계측학과 박사
관심분야 : 인공생명, 인공두뇌, 인공면역계, 자율분산시스템, 가상현실 등

심 귀 보 (Kwee-Bo Sim)

1984년 : 중앙대학교 전자공학과 학사
1986년 : 중앙대학교 전자공학과 학사
1990년 : The University of Tokyo 전자공학과 박사
1990년 : 동경대학 생산기술연구소 연구원
1998년~현재 : 한국 퍼지 및 지능시스템 학회 이사 및 논문지 편집 위원
1999년~현재 : 중앙대학교 전자전기공학부 교수
관심분야 : 인공생명, 진화연산, 지능로봇시스템, 뉴로-퍼지 및 소프트 컴퓨팅, 자율분산 시스템, 로봇비전, 진화하드웨어, 인공면역계 등