셀룰라 오토마타 신경망의 특성

A Characteristics of Cellular Automata Neural Systems

이 동 욱, 심 귀 보
로보틱스 및 지능정보 시스템 연구실
중앙대학교 공과대학 전자전기공학부
Tel : 02)820-5319, Fax : 02)817-0553, E-mail : {dwlee, kbsim}@cau.ac.kr

Dong-Wook Lee, Kwee-Bo Sim
Robotics & Intelligent Information System Lab.
School of Electrical & Electronic Eng., Chung-Ang Univ.
Tel : +82-2-820-5319, Fax : +82-2-817-0553, E-mail : {dwlee, kbsim}@cau.ac.kr

ABSTRACT

셀룰라 오토마타 신경망은 저자에 의하여 개발된 신경망으로써 주변의 셀과 국소적인 연결을 가지며 셀룰라 오토마타의 발생규칙에 따라 생성되는 신경망이다. 셀룰라 오토마타 신경망을 간단히 줄여서 ECANS라고 한다. 본 신경망은 카오스 뉴런 모델을 사용하며 뉴런사이의 연결강도는 흥분성 또는 억제성 결합을 갖는다. 신호의 전달방식은 펄스의 형태로서 뉴런이 발화하면 '1' 발화하지 않으면 '0'이 된다. 본 논문에서는 셀룰라 오토마타를 구성하는 요소별 특징을 살펴보고 주어진 문제에 적합한 셀룰라 오토마타 신경망을 얻어내기 위한 진화방법으로서 DNA 코딩방법을 제안한다. 제안한 방법의 유효성을 시뮬레이션을 통하여 검증한다.

Ⅰ. 서 론

최근, 인공생명, 신경망, 로봇과 관련된 학회에서 인공두뇌(Artificial Brain)를 개발하기 위한 초보적인 논문이 많이 발표되고 있다[1,3,4,9]. 이러한 연구들 중에는 인공두뇌를 개발하기 위한 접근방법으로 생물학적 두뇌 모델을 연구하여 이를 공학적으로 구현하기 위한 방법이 많다. 특히 기존의 신경망의 파라미터나 구조를 진화하던 단계에서 더 나아가 신경망의 합성규칙을 진화시키는 방법이 연구되고 있다. 이것은 생물체의 발생과 진화 메커니즘에서부터 힌트를 얻은 것으로서, 신경망의 크기가 커지더라도 진화의 복잡성이 증가하지 않는 장점이 있다. 따라서, 궁극적으로 인공 뇌와 같이 고도로 복잡한 시스템을 구축하는데 유용한 방법이 될 수 있다.

생물학적 발생을 모델링 한 도구로서 L-시스템, 셀룰라 오토마타 및 세포의 분열을 모델링 한 방법 등이 있다. Boers[1], Gruau[4] 등은 L-시스템 기반의 신경망의 설계방법을 제안하였고, Garis[3] 등은 셀룰라 오토마타 기반의 인공 뇌를 개발하기 위한 연구를 수행하고 있다. 본 논문에서는 셀룰라 오토마타와 진화 알고리즘을 기반으로 한 새로운 형태의 신경망 구성방법을 제안한다. 우리는 이것을 진화하는 셀룰라 오토마타 신경망(Evolving Cellular Automata Neural Systems)이라 하고 간단히 ECANS으로 명명하였다[6]. 셀룰라 오토마타는 발생규칙에 의해 셀이 국소적으로 발전하는 시스템이다.

여기서 우리는 발생규칙을 염색체로 사용한다. ECANS를 진화적으로 설계하기 위하여 우리는 기본적인 신경망의 동작방식을 바꿀 필요가 있었다. 따라서 연결강도를 없애고 그 대신 뉴런의 종류와 배열이 신경망의 역할에 중요한 역할이 되도록 하였다. 신경망은 초기셀의 배열 및 규칙에 따라 다양한 형태를 갖게 되는데 이 형태는 셀룰라 오토마타에 의해 결정된다. 셀룰라 오토마타에서 각각의 셀은 신경망의 뉴런에 셀의 상태는 뉴런의 종류에 대응된다. 이때 셀은 주변 셀과의 연결관계에 따라 여러 가지 모양을 갖는다. 이러한 특징을 얻기 위하여 각 뉴런은 기존의 시그모이드 뉴런에 비하여 복잡한 특성을 갖는 카오스 뉴런을 사용하였다. 카오스 뉴런은 실제 생물체의 뉴런에 가깝게 모델링 한 것으로서 Nagumo-Sato가 제안한 모델이다. 카오스 뉴런 모델에서 출력값을 0과 1을 사용함으로써 네트워크의 신호는 펄스의 형태로 전달되고 이 펄스의 밀도에 따라 출력의 세기를 나타낼 수 있다.

일반적으로 유전자 알고리즘에서 유전자의 배열은 곧 표현형의 구조를 직접 나타낸다. 이 경우 표현형이 매우 복잡한 구조를 가진다면 유전형의 길이도 매우 길어지는 단점을 가지게 된다. 따라서 본 논문에서는 유전형과 표현형의 변환 과정에 발생모델을 이용함으로써 적은 길이의 유전자형을 가지고 표현형을 나타낼 수 있는 방법을 이용한다. 이때 사용하는 발생모델은 셀룰라 오토마타이다. 이것은 실제 생물체의 생물체가 발생하는 과정에서 각각의 세포는 유전정보(발생규칙) 및 주변상태에 따라 발달하는 것

과 유사하다. 본 논문에서는 DNA 코딩 방법을 이용하여 셀룰라 오토마타의 규칙을 표현하는 방법을 제안한다. 기존에는 셀룰라 오토마타의 규칙을 진화하기 위하여 모든 상태가 고려되는 테이블을 이용하였다. 그러나 이 방법은 상태가 늘어남에 따라 염색체의 길이가 길어질 뿐만 아니라 유사한 상태에 대한 다음 상태의 결정에 대한 규칙성이 없다. 즉 모든 가능한 상태에 대하여 개별적인 다음 상태의 결정값을 갖는다. 본 논문에서 제안하는 셀룰라 오토마타 규칙의 DNA 코딩 방법은 이러한 문제점을 해결한다.

2절에서는 본 저자가 이미 개발한 셀룰라 오토마타 신경망의 구성요소에 대하여 설명하고 3절에서는 셀룰라 오토마타 신경망의 발생 규칙의 진화를 위한 DNA 코딩 방법을 제안하녀 4절에서는 본 신경망의 특징을 정리하고 5절에서 시뮬레이션 결과를 소개한다.

II. 셀룰라 오토마타 신경망

2.1 셀룰라 오토마타

셀룰라 오토마타[5,7]는 공간, 시간, 상태가 이산적인 동적 시스템이다. 셀룰라 오토마타에서 이산적인 양으로 분할된 공간을 셀이라 부르며 한 시간에 유한개의 상태 중의 한 가지를 가질 수 있다. 격자구조 속의 셀의 상태는 국소적인 규칙에 의하여 수정된다. 즉, 주어진 시간의 셀의 상태는 한 단계 전의 자기자신의 상태와 근처 주변 셀의 상태에만 의존한다. 또한 격자상의 모든 셀의 상태는 동시적으로 수정된다.

2.2 기본구성

제안된 시스템에서 셀의 상태는 주변 이웃셀과의 관계에 따라 정의된다. 기본적인 연결방식은 단지 인접한 셀들만 연결을 시키는 것이다. 연결은 흥분성 연결, 억제성 연결, 연결 없음 의 3가지 형태 중 하나를 가진다. 각 셀은 이웃의 셀 및 다음 상태의 자신의 셀과 연결할 수 있다. 따라서 총 연결할 수 있는 가짓수는 3^3=27이 된다. 셀룰라 오토마타에서 상태의 수는 시스템의 복잡도 및 규칙(rule)의 크기에 영향을 미친다. 본 논문에서는 연결의 대칭성을 고려해 그림 1과 같은 8가지의 셀을 사용하였다. 이와 같이 선택한 이유는 단순한 구성요소를 가지고도 복잡한 네트워크를 표현 가능하다고 판단했기 때문이다. 이처럼 셀의 가짓수를 줄임으로서 GA가 탐색할 상태공간이 줄어서 진화의 속도가 빨라지는 이점이 생긴다.

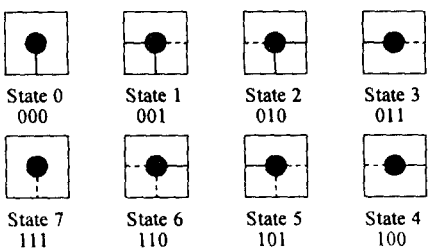


그림 1. 셀의 연결방식에 따른 셀의 상태
(실선 : 흥분성 연결, 점선 : 억제성 연결)

셀의 상태는 3절의 DNA 코딩 방법에서 논리 연산에 적용하기 위하여 비트 값을 사용한다. 따라서 억제성 연결과 흥분성 연결은 상호 보수적인 관계가 되도록 그림 1과 같이 셀의 상태를 설정하였다.

2.3 카오스 뉴런 모델

ECANS의 뉴런은 실제의 신경세포막에 충실하게 만들어진 Hodgkin-Huxley 방정식을 토대로 재구성한 Nagumo-Sato의 신경모델[13]을 사용하였다. 이 모델은 비교적 단순한 구조를 가지고 있음에도 불구하고 매우 복잡한 특성(입력값에 따라 셀의 내부상태가 카오스적인 현상을 나타냄)을 가진다. 본 논문에서, ECANS의 셀은 동적인 특성을 갖는 카오스 뉴런으로 모델링 하였다. Nagumo-Sato의 카오스 뉴런의 동작 방정식은 다음과 같다.

$$y(t+1) = u(x(t+1)) \tag{1}$$

$$x(t+1) = I(t) - \alpha \sum_{d=0}^{t} k^d y(t-d) - \theta \tag{2}$$

단, $y(t)$는 시간 t에서의 출력, $x(t)$는 내부 상태, $I(t)$는 입력, $u()$는 단위 계단 함수, k^d는 0에서 1사이의 값을 가지는 감쇠계수, 상수 α는 양의 파라미터이며 θ는 뉴런의 문턱 값이다.

위의 (1)식에서 출력은 on(1)/off(0)의 특성을 갖게 된다. 따라서 실제의 적용문제에서 신경망이 동작하기 위한 클럭(clock)이 필요하다. 본 논문에서는 0.02초의 주기를 갖는 클럭을 사용하였다. 셀룰라 오토마타 신경망에서 한 뉴런은 오직 주변(상하좌우)의 뉴런과 연결을 갖으며 연결강도는 -1, 0, 1중 하나를 갖는다.

2.4 펄스형 동작방식

위의 방정식에 의하여 한 뉴런의 출력은 입력에 비례하는 펄스밀도를 갖는 신호를 출력한다(뉴런이 발화할 때 펄스 생성). 이때 뉴런의 카오스적인 특성으로 인하여 펄스의 배열은 비주기적이며 랜덤한 성질을 갖는다. 이것은 네트워크가 정상적으로 동작하도록 하는 역할을 한다. 그 이유는 만약 이웃하는 뉴런에 갖은 신호가 들어왔을 경우 두 뉴런이 같은 주기의 펄스를 내보낸다면 위상차에 의해서 같은 경우라도 신호가 증폭되거나 소멸하는 등 다른 결과가 나타날 수 있다. 그러나 카오스 뉴런은 과거 자신의 정보를 이용하고 펄스의 발생이 비주기적이기 때문에 신호의 증폭이나 소멸 등의 현상은 일어나지 않는다. ECANS에서 한 뉴런은 오직 이웃하는 뉴런과 연결을 갖으며 이때 연결강도는 -1, 0, 1중 하나를 갖는다. 즉 연결은 단지 흥분성인가 억제성 연결인가만 구분된다. 그림 2는 펄스형태로 동작하는 생물학적 뉴런의 모델을 나타낸다.

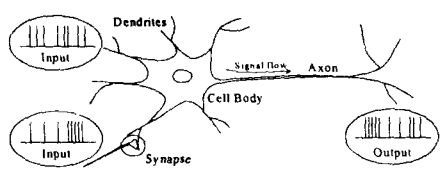


그림 2. 펄스형 동작 방식을 가지는 생물학적 뉴런 모델
뉴런의 최종 출력은 펄스의 밀도에 의해 계산된다.

2.5 네트워크의 카오스 특성

셀룰라 오토마타 신경망의 뉴런 하나는 모두 카오스적인 다이나믹스를 가지고 있다(그림 4). 또한 몇 개의 뉴런을 연결한 것도 역시 카오스 현상을 나타낸다. 그림 3은 간단한 셀룰라 오토마타 신경망을 나타내며 그림 4는 그림 3의 신경망에 대한 출력 뉴런의 상태인 x_1에서 x_4의 값을 각각 나타낸다. 이 그림은 입력 i의 변화에 따른 각 네트워크의 상태 x에 대한 분기도(bifurcation diagram)를 나타낸다. ECANS에서 뉴런간에 연결강도는 특별히 없으며, 오직 흥분성 결합(1), 억제성 결합(-1), 무결합의 상태만 가진다. 따라서 네트워크는 내우 단순하게 구성된다. 그러나 뉴런들의 카오스 특성 때문에 네트워크 전체는 매우 복잡한 행동을 보인다.

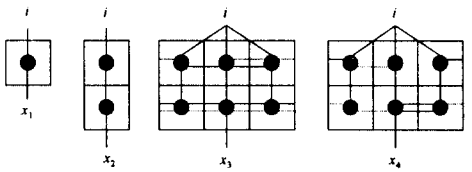
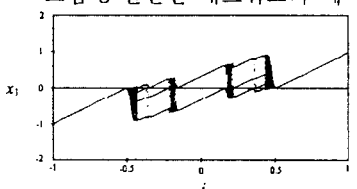
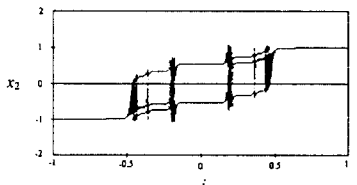


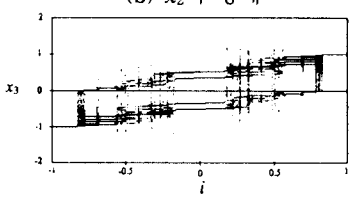
그림 3 간단한 네트워크의 예



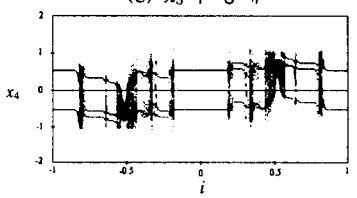
(a) x_1의 상태



(b) x_2의 상태



(c) x_3의 상태



(d) x_4의 상태

그림 4. 그림 3에서 나타낸 네트워크의 입력 i에 따른 상태 x의 분기도

Ⅲ. 셀룰라 오토마타 신경망의 진화

3.1 신경망의 발생 및 진화

생물체의 대표적인 세 가지 자기조직화 현상[8]은 발생(Ontogeny), 진화(Phylogeny), 학습(Epigenesis)이다. 이런 자기조직화 현상이 일어나기 위해서는 현상의 다양성이 최대화되는 카오스의 경계(edge of chaos)[5]를 나타내는 물리적인 조건이 필요하다. 기본적으로 진화는 개체들의 조건을 카오스의 경계로 유도하는 역할을 하며 한번 카오스의 경계 영역을 확보한 상태에서는 진화 효율이 매우 높아진다.

그림 5는 초기 셀의 발생 과정 및 입출력 뉴런을 네트워크에 어떻게 연결하는지를 보여주는 그림이다. 입력 및 출력은 그림 5의 최종 네트워크의 그림과 같이 교대로 네트워크에 인가된다.

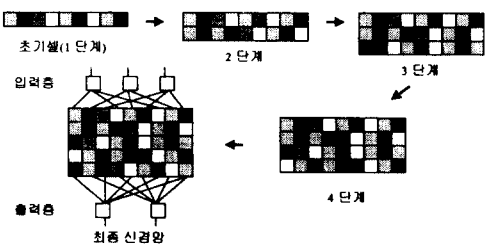


그림 5. ECANS의 발생단계

3.2 DNA 코딩에 의한 CA 규칙의 진화

3.2.1 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[10]. 또한 염기 3개의 배열이 한 의미단위를 이루어 해석된다.

	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

아미노산 약어 알라닌-Ala, 아르기닌-Arg, 아스파라긴-Asn, 아스파르트산-Asp, 시스테인-Cys, 글루탐산-Glu, 글루타민-Gln, 글리신-Gly, 히스티딘-His, 이소루신-Ile, 루신-Leu, 리신-Lys, 메티오닌-Met, 페닐알라닌-Phe, 프롤린-Pro, 세린-Ser, 트레오닌-Thr, 트립토판-Trp, 티로신-Tyr, 발린-Val.

표 1. RNA(DNA) 코돈과 생성하는 아미노산

* DNA에서는 U대신 T를 사용한다.

이 의미단위를 생물학적인 용어로 코돈(codon)이라 한다. 코돈의 가지수는 4×4×4 = 64개이며 이것이 코드화하는 아미노산은 20가지이다. 코돈의 64가지 패턴에 대하여 생성하는 아미노산이 20가지인 이유는 다른 코돈이 같은 아미노산을 만들기도 하기 때문이다. 이것은 표 1에 나타나 있다.

DNA는 RNA로 전사되어 리보솜에서 단백질로 번역된다. 즉 아미노산을 암호화하는 DNA의 배열에 따라 아미노산의 합성순서를 결정하여 여러 종류의 단백질을 만들어낸다. RNA의 단백질로의 번역은 AUG에서 시작해서 UGA에서 번역이 끝난다.

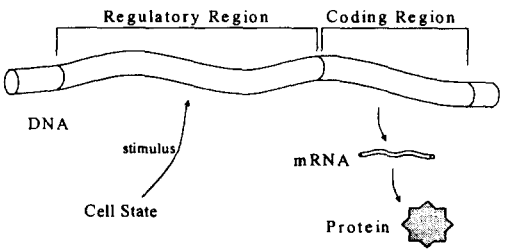


그림 6. 유전자 구조의 일반적인 구조

생물학적 유전자의 기본적인 구조는 그림 6과 같이 되어있다. 단백질을 직접 코드화 하는 부위와 그 코드 부위의 발현을 조절하는 조절부위로 구성되어 있다. 코드화 부위는 조절부위의 명령에 의해 세포내의 조건이 만족할 때 단백질로 번역된다. 이것은 발생모델의 규칙의 표현방식과 유사하다. 즉, 조절부위는 규칙의 전건부 또는 세포의 현재 주변상태, 코드화 부위는 후건부 또는 세포의 다음 상태에 대응된다.

3.2.2 DNA 코딩 방법의 특징

DNA의 동작을 모방한 코딩방법에 대한 연구는 Yoshikawa [12] 및 Deaton[2] 등에 의하여 기본적인 방법이 연구되었다. 염색체는 기본적으로 4가지 염기의 배열로 이루어져 있고 아미노산을 번역하는 것과 마찬가지로 코돈 단위로 번역한다. 또한 그림 7와 같이 유전자의 번역 시작점이 일정하지 않기 때문에 중복 번역을 허용하며, 때에 따라서는 번역되지 않는 부분도 존재한다. 교차점도 임으로 주어지기 때문에 염색체의 길이도 가변적이다. Yoshikawa의 방법은 Decoding 테이블을 만듦으로서 규칙의 표현에 적합하게 구성되어 있다. 이상의 특징을 정의하면 아래와 같은 4가지로 정리할 수 있다.

◆ DNA 코딩 방법의 특징
(1) 지식의 융통성 있는 표현이 가능하다.
(2) 코딩에 여분과 중복이 있다.
(3) 염색체의 길이가 가변적이다.
(4) 교차점이 강제적으로 주어지지 않는다.

CGATGCATTGCCACTTGCATGCGTCGACGCTGTGACGTATG
ACT

His Cys His Leu His Ala Ser Thr ···

그림 7. 염색체의 번역 및 유전자의 중복

Wu[11] 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) 성질을 모두 가지고 있다. 유용한 스키마를 찾는 능력에 대한 일반 GA방법(고정위치 표현법)과의 비교에서 그는 다음과 같은 결론을 얻어냈다. (1) 염색체의 길이가 짧을 경우에는 일반적인 고정위치 표현방법(fixed representation)이 우수했으나 염색체의 길이가 길수록 가변위치 표현법의 성능이 더 좋아진다. (2) 가변위치 표현법에서 개체군의 다양성이 더욱 높다. 따라서 개체의 다양성을 최대로 유지하며 탐색을 수행한다.

3.2.3 DNA 코딩 방법에 의한 CA 규칙의 번역

셀룰라 오토마타의 규칙을 코드화하고 번역하기 위해서는 표 1의 아미노산의 코드표와 같은 번역(decoding)표가 필요하다. 우선 셀룰라 오토마타의 규칙을 표현하기 위한 방법을 결정하고 이에 따라 필요한 표를 작성해 나간다.

1차원 셀룰라 오토마타에서 셀의 이웃반경을 r이라 했을 때 규칙($\Phi_i(\sigma_i)$)을 나타내는 일반적인 표현은 (3)식과 같다.

$$\sigma_i^{new} = \Phi_i(\sigma_i) = \phi(\sigma_{i-r}, \cdots, \sigma_{i+r}) \tag{3}$$

이웃반경 r이 1인 경우를 생각할 때 j 단계의 i 번째 셀의 상태와 그 이웃상태에 따라 j+1 단계의 i 셀의 상태는 결국 σ_{i-1}^j, σ_i^j, $\sigma_{i+1}^j \rightarrow \sigma_i^{j+1}$의 함수이다. 또한 함수 ϕ가 n 개의 부분 규칙함수로 이루어져 있다고 가정하면 (4)식과 같이 표현할 수 있다.

$$\phi = \begin{cases} \phi_1, & \text{조건1} \\ \phi_2, & \text{조건2} \\ \cdots & \\ \phi_n, & \text{조건}n \end{cases} \tag{4}$$

단, n은 최대 (총 상태)$^{2r+1}$ 의 값을 가질 수 있다.

(4)식에서 조건1 ~ 조건n은 결국 현재의 이웃에 의하여 결정되는 식이므로 본 논문에서는 다음의 (5)~(7)식을 사용하였다.

$$\text{If } \sigma_{i-1}^j = s_1 \text{ and } \sigma_i^j = s_2 \text{ and } \sigma_{i+1}^j = s_3 \tag{5}$$

$$\text{If } \sigma_l = s_1 \text{ and } \sigma_m = s_2 \tag{6}$$

$$\text{If } \sigma_n = s_1 \tag{7}$$

단, s_1, s_2, s_3는 셀의 상태를 나타내며 $l, m, n \in \{i-1, i, i+1\}$, $l \neq m$ 이다.

여기에서 (3)식을 조건으로 사용한 규칙은 현재 이웃하는 셀의 상태가 모두 결정된 상태로 매우 세부적이며 특정한 규칙이 되고, (4)식과 (5)식의 조건에 의해 결정된 규칙은 여러 가지의 상태를 포함하는 포괄적인 규칙이 된다. 이때 부분규칙 ϕ_k (단,

$k=\{0,1,\cdots n\}$)는 다음에 정의하는 연산자(operation)를 사용하여 구성된다.

◆ 연산자(operation)의 종류

AND, OR, XOR, ADD, DIFF, NOT, 3XOR, State

표 2. DNA 코돈의 해석순서

(case 1) 1 state	(case 2) 2 state	(case 3) 3 state
① 시작 코돈		
② Order		
③ State		
④′ A/B (B)	④ A/B (A)	
(⑤′⑥″생략)	⑤ State	
	⑥′ A/B (B)	⑥ A/B (A)
⑦′ Operation		⑦ State
⑧″ Next State	⑧″ Order	⑧ Next State

조건 판단 및 연산의 순서를 결정하기 위하여 세 가지 셀의 순서를 나타내는 Order(총 6가지)를 도입하였고 몇 개의 상태를 고려할 것인 가를 And/Break(A/B)를 사용하여 제어하였다. 즉, A/B의 값에 따라 ④번과 ⑥번에서 흐름이 바뀐다. case 3의 경우 현재 이웃하는 세 가지 셀의 상태를 모두 조건으로 고려하는 경우이고 case 1과 case 2는 각각 세 셀 중 한 가지 및 두 가지의 셀을 조건으로 고려하는 경우이다. 연산(operation)은 기본적으로 두 가지에 셀의 상태를 가지고 수행하나 NOT은 한가지 셀, 3XOR은 세 가지 셀의 상태를 가지고 연산한다. 또한 Operation이 State인 경우 다음 코돈을 해석하여 특정 상태를 선택한다.

표 3. 코돈의 해석 표

Amino Acid	Phe	Leu	Val	Ser	Pro	Thr	Ala	Tyr	His	Gln	Asn	Lys	Asp	Glu	Cys	Arg	Gly
Order	0	2	1	3	4	1	5	1	3	1	5	4	4	2	5	0	4
State	0	0	1	3	7	2	7	4	5	5	2	5	1	6	3	4	6
A/B	A	A	A	B	A	A	A	A	A	A	A	A	A	A	B	B	B
Operation	0	0	1	3	2	2	1	4	5	5	5	5	6	6	3	4	6

* Ile, Met, Trp 및 종료 코돈은 번역 시작 기호로 사용하기 위해 비워둠

표 3은 DNA 코돈을 해석하기 위하여 각 코돈에 대하여 Order, State, A/B, Operation을 대응시킨 표이다. 코돈의 배열 순서에 따라 표 2 및 표 3을 참조하면 셀룰라 오토마타 규칙이 생성된다. 표 3은 하나의 코돈이 돌연변이를 일으켰을 때 다른 어떠한 종류의 상태로 바뀔 수 있도록 표를 배열하였다. 생물체에서 만들어지는 단백질은 일반적으로 최소 55개에서 많게는 수백 개의 아미노산으로 구성되어 있다. 따라서 한 개의 시작 코돈만 가지고도 충분히 유전적인 암호를 코드화 할 수 있다. 그러나 하나의 셀룰라 오토마타 부분규칙은 6~8개의 코돈으로 구성되므로 1/64의 비율로 나타나는 하나의 시작 코돈을 가지고 진화를 하는 것은 염색체를 상당히 비효율적으로 사용하는 것이 된다. 따라서 본 실험에서는 4개의 코돈을 시작 코돈으로 사용하였다. 규칙의 길이와 시작 코돈의 발생빈도의 관계는 더 연구해야 할 과제이다.

그림 4의 첫 번째 규칙을 해석하면 다음과 같다. 우선 시작 코돈인 ATG에서 번역이 시작되고 차례로 3개씩 표 2를 따라 다음과 같이 번역된다.

① ATG : 시작코돈 ② His : Order(2-1-3) ③ Cys : State(3) ④ His : A/B(A) ⑤ Leu : State(0) ⑥ His : A/B(A) ⑦ Ala : State(7) ⑧ Ser : Next State(3)

이것을 규칙으로 표현하면 If $\sigma_{i-1}^{j}=0$ and $\sigma_{i}^{j}=3$ and $\sigma_{i+1}^{j}=7$ then $\sigma_{i}^{j+1}=3$ 과 같이 된다.

Ⅳ. 셀룰라 오토마타 신경망의 특성

본 논문에서 제안하는 셀룰라 오토마타 신경망의 특성을 살펴보면 다음과 같다.

신경망의 구성요소

(1) 뉴런의 특성방정식으로 카오스 뉴런 모델을 사용한다.

(2) 각 뉴런은 pulse(on/off)에 의해 동작한다.

신경망의 구조

(3) 각 뉴런은 이웃하는 뉴런과 연결을 갖는다.

(4) 연결강도가 아닌 연결방식에 의하여 특성을 나타낸다.

신경망의 생성방법

(5) 신경망의 발생은 셀룰라 오토마타의 규칙을 따른다.(발생규칙은 DNA 코딩방법으로 표현됨)

(6) 진화 방식에 의하여 기능을 획득한다.

Ⅴ. 시뮬레이션 결과

본 논문에서는 제안한 방식의 신경망의 유효성을 검증하기 위하여 간단한 로봇의 주행문제에 적용하였다. 그림 4는 50세대 진화 후 얻어진 로봇의 궤적을 나타낸다. 그림 5는 로봇의 센서 입력값과 바퀴의 출력값이다. 시뮬레이션을 위한 조건을 다음과 같이 설정하였다.

- 셀룰라 오토마타 신경망 파라미터
  - 입력 뉴런 : 5(로봇의 센서수)
  - 출력 뉴런 : 2(바퀴의 개수)
  - 은닉층 네트워크의 크기 : 10 × 5
- 진화 파라미터
  - 개체군의 크기 : 100
  - 초기 염색체의 길이 : 1024
  - 교차 확률 : 0.8
  - 돌연변이 확률 : 0.1

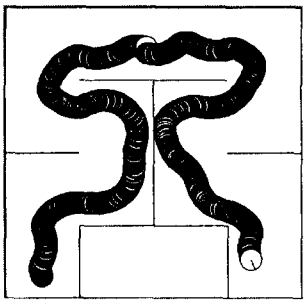


Fig. 4 Trace of Best Individual(Robot)

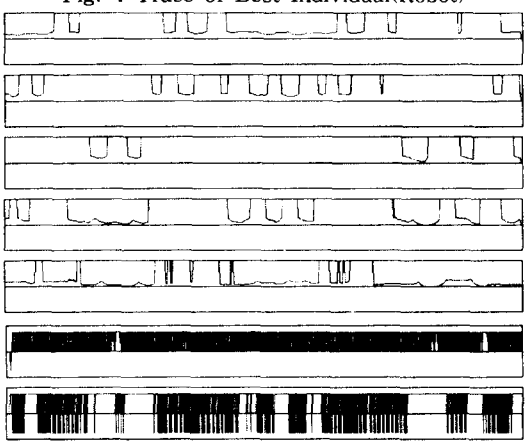


Fig. 5 Input(5) / Output Data(2)

진화를 통하여 최상의 개체는 약 50세대 후에 얻어졌으며 염색체의 길이는 1436이었다. 초기 길이에 비하여 약간 늘어났으며 번역된 규칙의 수 또한 초기세대 62개에서 120개로 늘어났다. 이를 통하여 규칙을 나타내는 시작 코돈의 밀도가 높아졌음을 알 수 있다.

Ⅵ. 결 론

본 논문에서는 셀룰라 오토마타 신경망의 구성 및 진화방법을 제안하였다. 뉴런을 하나의 셀로 여기고 이것을 CA의 규칙에 따라 발생하여 신경망을 구성하였다. 신경망의 작동을 연결강도가 아닌 연결방식으로 바꾸기 위하여 카오스 뉴런을 도입하였고 신호의 세기는 펄스 밀도로서 표현되도록 하였다. 본 신경망은 CA의 규칙에 따라서 다양한 형태를 갖게된다. 본 논문에서는 문제에 적합한 신경망을 발생하는 규칙을 찾기 위하여 규칙을 염색체로 코드화 하여 진화를 시켰다. 일반적으로 규칙을 직접 코드화 하면 염색체의 길이가 매우 길어져 간단한 문제를 진화시키는 데에도 시간이 오래 걸리게 된다. 본 논문에서는 이러한 문제점을 개선하기 위하여 효율적인 규칙의 진화를 위한 DNA 코딩 방법을 제안하였다. 규칙을 일반적인 규칙(case1, case2)과 세부적인 규칙(case 3)으로 나누어 적은 수의 규칙으로 다양한 표현이 가능하도록 하였다. 제안된 방법에 의하여 주어진 목적을 수행할 수 있는 신경망 제어기가 구성되는 가를 간단한 시뮬레이션을 통하여 알아보았다.

인공생명은 기본적인 구성요소 및 법칙을 가지고 생물학적 현상을 합성하는 상향식 접근방식을 취한다. 반면 기존의 생물학은 생물의 특성을 분석하여 그 원리를 밝혀 내는 하향식 접근방법을 이용한다. 본 논문에서 제안한 신경망은 인공생명의 상향식 접근방법에 의하여 설계되었다. 즉 신경망의 구성요소의 특징을 설정하고 이것을 발생 및 진화를 통하여 이 구성요소의 상호작용에 의한 전체적인 유용한 특성이 나타나도록 하였다. 앞으로는 이와 같이 설계된 신경망을 분석하여 더 발전된 구조를 설계하는데 이용할 예정이다. 본 신경망은 FPGA를 이용하여 구현하는 문제도 연구중이다.

감사의글

본 연구는 과학재단 특정기초 연구비 (96-0102-13-01-3) 지원으로 수행되었으며 지원에 감사를 드립니다.

참고 문헌

[1] E.J.W. Boers, H. Kuiper, B.L.M. Happel, and S. Kuyper, "Designing Modular Artificial Neural Networks," *Proceedings of Computer Science in the Netherlands*, pp. 87-96, 1993.

[2] R. Deaton et. al, "A DNA Based Implementation of an Evolutionary Search for Good Encodings for DNA Computation," *Proceedings of '97 Magnetics Conference*, pp. 267- 271, 1997.

[3] Hugo de Garis, "CAM-BRAIN : The Genetic Program- ming of an Artificial Brain Which Grows/Evolves at Electronic Speeds in a Cellular Automata Machine," *Proceedings of The First International Conference on Evolutionary Computation*, vol. 1, pp. 337-339b, 1994.

[4] F. Gruau, D. Whitley, "Adding Learning to the Cellular Development of Neural Networks: Evolution and the Baldwin Effect," *Evolutionary Computation*, vol. 1-3, pp. 213-233, 1993

[5] C.G. Langton, "Life at the Edge of Chaos," *Artificial Life II*, Addison-Wesley, pp. 41-91, 1992.

[6] D.W. Lee, K.B. Sim, "Evolving Cellular Automata Neural Systems 1(ECANS1)," *Proceeding of The 3rd Asian Fuzzy System Symposium*, pp. 158-163, 1998. 6.

[7] M. Sipper, "Non-Uniform Cellular Automata : Evaluation in Rule Space and Formation of Complex Structures," *Artificial Life VI*, The MIT Press, pp. 394-399, 1994.

[8] M. Sipper et al. "A Phylogenetic, Ontogenetic, and Epigenetic View of Bio-Inspired Hardware Systems," *IEEE Transactions on Evolutionary Computation*, vol. 1 no. 1, pp. 83-97, 1997. 4.

[9] M. Sugisaka, "Design of an artificial brain for robots," *Proceedings of The Third International Symposium on Artificial Life and Robotics*, pp. (I -2) - (I -11), 1998.

[10] R.A. Wallace, G.P. Sanders, R.J. Ferl, *BIOLOGY : The Science of Life 3rd eds.*, HarperCollins Publishers Inc., 1991.(이광웅 외

공역, *생물학*, 을유문화사, 1993.)
[11] 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.
[12] T. Yomohiro, T. Furuhashi, Y. Uchikawa, "A Combination of DNA Coding Method with Pseudo-Bacterial GA for Acquisition of Fuzzy Control Rules," *Proceedings of 1st Online Workshop on Soft Computing*, Aug. 19-30, 1996.
[13] 合原一幸 編著, *ニュラルシステムにおけるカオス*, 東京電氣大學出版局, 1993.(정호선, 여진경 공역, *뇌와 카오스*, Ohm사, 1994.)