1999 年 度 大 韓 電 子 工 學 會 夏 季 綜 合 學 術 大 會 論 文 集 第 22 卷 第 1 號 99/6

발생모델의 진화를 위한 DNA 코딩방법

심 귀 보(沈 貴 寶), 이 동 욱(李 東 昱)
중앙대학교 전자전기공학부
전화 : 02-820-5319 / 팩스 : 02-817-0553

DNA Coding Method for Evolution of Developmental Model

Kwee-Bo Sim, Dong-Wook Lee
School of Electrical and Electronic Engineering, Chung-Ang University
E-mail : kbsim@cau.ac.kr, dwlee@ms.cau.ac.kr

Abstract

Rapid progress in the modeling of biological structures and simulation of their development has occurred over the last few years. Cellular automata (CA) and Lindenmayer-system(L-system) are the representative models of development/morphogenesis of multicellular organism. L-system is applied to the visualization of biological plant. Also, CA are applied to the study of artificial life and to the construction of an artificial brain. To design the L-system and CA automatically, we make this model evolve. It is necessary to code the developmental rules for evolution. In this paper, we propose a DNA coding method for evolution the models of development/ morphogenesis of biological multicellular organisms. DNA coding has the redundancy and overlapping of gene and is apt for the representation of the rule. In this paper, we propose the DNA coding method of CA and L-system.

I. 서론

생물의 3가지 자기조직화 현상을 모방한 인공생명 모델은 각각 진화모델, 학습모델, 발생/발달 모델이다. 이 세가지 모델은 인공생명 연구에 가장 중추적인 역할을 맡고 있는 것으로서 진화모델에는 진화 알고리즘이 있고 학습모델로는 신경망, 강화학습, 인공면역계 등이 있으며 발생/발달 모델로는 셀룰라 오토마타(cellular automata, 이하 CA)[1], 린드마이어 시스템(L- system, 이하 L-시스템)[2] 등이 있다. 이중 진화 및 학습 알고리즘은 유전자 알고리즘(GA)과 신경망이 일반화되어 공학의 전반적인 분야에 적용이 되고 있다. 발생모델 중 CA는 인공생명의 연구모델 또는 인공두뇌의 건축 방법으로, L-시스템은 컴퓨터 그래픽스 등에 적용되고 있다.

발생모델은 일반적으로 초기 구조 및 생성규칙으로 구성된다. 즉 초기 구조로부터 규칙에 의해 발전시켜 나가는 방식이다. 현재까지는 대부분 원하는 목적을 이루기 위하여 CA 및 L-시스템의 생성규칙을 사람이 직접 설계하였다. 특히 L-시스템은 원하는 모양을 얻기 위하여 많은 설계방법과 결과들이 제안되어왔다. 그러나 만약 더 복잡한 시스템을 구축하려고 하거나 이 모델들을 인공두뇌(또는 신경망) 등의 설계에 이용하기 위해서는 사람이 설계하는 방식으로는 한계가 발생하며 비효율적이다. 최근의 연구에서 발생모델은 GA 등으로 진화하려는 연구가 등장하고 있지만 염색체의 구조가 제한적이어서 일부의 간단한 문제에만 성공을 거두고 있다. 본 논문에서는 발생모델의 규칙을 진화하기 위한 유연한 구조의 코딩방법을 제안한다. 이 방법은 생물학적 DNA의 구조를 모방한 것으로서 DNA 코딩이라 명명되었다. DNA 코딩은 Yomohiro[3] 등에 의해서 퍼지 규칙을 코드와 하는데 적용된 방법이다. DNA 코딩은 DNA 염기배열이 단백질로 번역되는 것과 유사한 과정을 통하여 규칙으로 번역된다. DNA 코딩을 이용한 진화 알고리즘에 대한 연구는 아직 초기 상태이지만 점차로 이론적인 논문과 연구가 이루어지고 있는 실정이다. 본 논문에서는 CA와 L-시스템의 DNA 코딩방법을 제안한다.

II. 발생/발달 모델

2.1 셀룰라 오토마타(CA)

CA는 von Neumann에 의하여 처음으로 고안된 이산적인 동적 시스템이다. CA의 공간은 '셀'이라는 이산된 블록으로 나누어져 있으며 시간의 경과에 따라 이산적인 단계로 발전한다. 이때 셀의 상태는 국소적인 규칙에 의하여 갱신된다. 즉, 한 시간의 셀의 상태는 한 단계 이전의 자신의 상태와 한 단계 이전의 이웃하는 셀의 상태에 의하여 결정된다. 또한 격자상의 모든 셀은 이산적인 시간단계로 동시적으로 진행된다.

d-차원 CA는 격자공간(Z = 정수, 양과 음의 방향 모두 무한하다)에서 수행된다. 이때 Σ는 유한한 k(상태) 원소의 집합이라고 하면 CA의 다이나믹스는 전역적 함수에 의하여 다음과 같이 정의된다.

$$\Phi : \Sigma^Z \rightarrow \Sigma^Z \quad (1)$$

예를 들어 1차원 CA의 경우 국부적 수정함수 ϕ 는 한정된 영역에서 다음과 같이 정의된다.

$$\phi : \Sigma^{2r+1} \rightarrow \Sigma \quad (2)$$

단, r은 반경이다.

가장 중요한 CA의 성질은 이 함수가 유한한 룩업(lookup) 테이블로 결정된다는 것이다. 따라서 ϕ의 영역과 범위는 유한하다.

전역적 함수인 Φ는 정의에 의해서 ϕ로부터 (3)식과 같이 나타낼 수 있다.

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

하나의 예로서 k = 2, r = 1 인 경우, 0과 1로 구성된 비트열에서 각각의 위치는 두 개의 이웃에 의하여 논리적으로 갱신된다.

2.2 L-시스템

L-시스템[2]은 일종의 프랙탈로서 특별히 식물의 생장과정을 모델링 하기 위하여 개발한 수학적인 모델이다. L-시스템은 기본적으로 병렬적인 문자열의 재적용 메커니즘이다. 초기 문자열(axiom)로부터 생성 규칙의 반복적인 적용에 의하여 생성된 최종 문자열은 심벌(symbol)의 문맥에 따라 여러 가지 방식으로 해석된다. 가장 흔한 적용방식은 선을 그리는 방식을 결정하여 나무모양을 만들어 내는 것이다.

L-시스템에서 사용되는 용어들은 다음과 같이 정의된다.

- 문자(Alphabet) : 변수 또는 심벌
- Axiom : 초기 문자열
- 생성규칙(production rule) : syntax, 바꿔쓰기 규칙

언어(language)로서 L-시스템의 문법(Grammar) G는 (4)와 같이 표현된다.

$$G = \{ \Sigma, \Pi, \alpha \} \quad (4)$$

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

L-시스템은 생성 규칙의 문맥에 따라 문맥 자유 L-시스템과 문맥 의존 L-시스템으로 나뉘며, 그 외에 Bracket L-시스템, 파라미터 L-시스템, 지도형(map) L-시스템 등이 있다. L-시스템을 신경망의 생성에 이용하면 반복적이고 재귀적인 과정에 의하여 자기 유사성을 갖는 모듈형의 구조적인 특성을 도입할 수 있다. 최근 L-시스템에 의한 신경망의 설계방법도 많이 연구되고 있다. 나무 모양의 신경망, 그래프를 생성하는 G2L-시스템과 모듈형 신경망 등이 제안되어져 있다. 현재 L-시스템을 이용한 신경망은 매우 단순한 수준이지만 앞으로 뇌와 같이 큰 규모의 모듈형 신경망을 생성하는 매우 유용한 방법이 개발될 것으로 전망하고 있다.

Ⅲ. DNA 코딩방법

3.1 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 가지 수는 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.

표 1. RNA(DNA) 코돈과 생성하는 아미노산[4].

	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.

* DNA에서는 U대신 T를 사용한다.

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.

3.2 DNA 코딩방법

DNA 코딩방법에 대한 연구는 Yomohiro 등에 의하여 기본적인 방법이 연구되었다. 한편 Wu[5] 등은 GA에서의 가변위치 표현법(floating representation)에 대한 스키마 분석을 통하여 유효성을 증명하였다. DNA 코딩도 일종의 가변위치 표현법인데, 이 논문에서 가변위치 표현법은 고정위치 표현법에 비하여 염색체의 길이가 길수록 성능이 더 좋아지며 개체군의 다양성도 높다는 것을 실험적으로 증명하였다. 따라서 매우 복잡한 문제에 대하여 적은 수의 개체군을 가지고 개체의 다양성을 최대로 유지하며 탐색을 수행한다.

3.3 CA 규칙의 DNA 코딩

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

1차원 CA에서 셀의 이웃반경을 r이라 했을 때 규칙($\Phi_i(\sigma_i)$)을 나타내는 일반적인 표현은 (3)식과 같다. 이웃반경(r)이 1인 경우를 생각할 때 j 단계의 i 번째 셀의 상태와 그 이웃 상태에 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 부분규칙 함수로 이루어져 있다고 가정하면 (3)식의 ϕ는 (5)식과 같이 표현할 수 있다.

$$\phi = \begin{cases} \phi_1, & \text{조건1} \\ \phi_2, & \text{조건2} \\ \cdots & \\ \phi_n, & \text{조건}n \end{cases} \quad (5)$$

단, n은 최대 (총 상태)$^{2r+1}$의 값을 가질 수 있다.

(5)식에서 조건1 ~ 조건 n은 CA에서 현재의 이웃에 의하여 결정되는 식이다. 본 논문에서는 조건(전건부)을 어떤 이웃 셀의 상태를 고려할 것인가에 따라 (6)~(8)식의 3가지 패턴으로 나누었다.

Type 1: $\sigma_n = s_1$ (6)

Type 2: $\sigma_l = s_1$ and $\sigma_m = s_2$ (7)

Type 3: $\sigma^j_{i-1} = s_1$ and $\sigma^j_i = s_2$ and $\sigma^j_{i+1} = s_3$ (8)

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

여기에서 (8)식을 조건으로 사용한 규칙은 현재 이웃하는 셀의 상태를 모두 고려하는 상태로 매우 세부적이며 특정한 규칙이 되며 이것을 type 3 규칙으로 나타낸다. 또한 (6)식과 (7)식의 조건에 의해 결정된 규칙은 여러 가지의 상태를 포함하는 포괄적인 규칙으로 각각 type 1 및 type 2 규칙이 된다. 이때 다음상태를 결정하는 부분규칙(후건부) ϕ_k는 다음에 정의하는 논리 연산자(operator)를 사용하여 구성된다.

표 2. ϕ_k에서 사용되는 비트 논리 연산자.

No.	Operator	# of operand	No.	Operator	# of operand
0	3XOR	3	4	DIFFERENCE	2
1	AND	2	5	XOR	2
2	OR	2	6	NOT	1
3	ADD	2	7	State	1

DNA 코드는 3개 단위의 코돈으로 해석되며 표 2~4를 참조하여 그림 1의 흐름에 따라 번역된다. 전건부에서 규칙의 상태를 고려하고 후건부에서 피연산자의 순서를 결정하기 위한 셀의 순서를 나타내는 Order(표 4)를 도입하였고 A/B(And/Break)를 이용해 번역의 흐름을 제어하였다. 즉 A/B의 값에 따라 전건부에서 몇 개의 상태를 고려할 것인가가 결정된다. Type 3 규칙이 나타나기 위해서는 A/B에서 모두 A(And)가 나와야 하며 type 1의 규칙은 첫 번째 A/B에서 B가 나와야 한다. 후건부의 연산(operation)은 기본적으로 두 가지에 셀의 상태를 가지고 수행하나 NOT은 하나의 상태, 3XOR은 세 가지 상태를 이용한다. 또한 연산자(Operator)에서 State가 선택된 경우 다음 코돈을 해석하여 바로 다음상태를 결정한다. 이와 같이 CA의 규칙을 표현함으로써 기존의 table 표현방식에 비해 인접한 셀간에는 공통된 성질을 갖게 할 수 있으며 적은 수의 규칙을 가지고 많은 상태를 나타낼 수 있다.

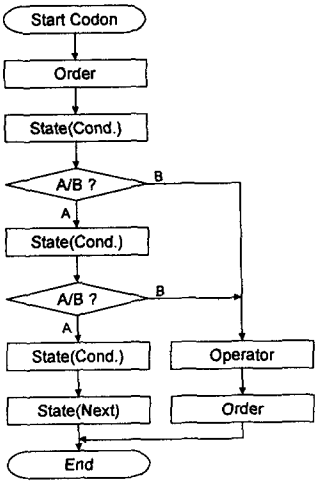

그림 1. DNA 코드의 번역 순서(각 불록은 코돈임)

표 3. CA의 코돈(아미노산)의 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
Op.	0	0	1	3	7	2	7	4	5	5	2	5	1	6	3	4	6

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

표 4. 첨자의 순서

Order	Index	Order	Index
0	i-1 → i → i+1	3	i → i+1 → i-1
1	i-1 → i+1 → i	4	i+1 → i-1 → i
2	i → i-1 → i+1	5	i+1 → i → i-1

표 3은 DNA 코돈을 해석하기 위하여 각 코돈에 대하여 Order, State, A/B, Operation을 대응시킨 표이다. 또한 표 4는 규칙에서 상태(state)를 고려할 때 순서(order)를 나타내는 표이다. 코돈의 배열 순서에 따라 그림 1과 표 1~4을 참조하면 CA의 부분규칙이 생성된다. 표 3 설계하는 규칙은 특별히 없으나 생물학적인 연구[6]에 따르면 하나의 코돈이 돌연변이를 일으켰을 때 다른 어떠한 종류의 상태로 바뀔 수 있도록 (changeability)설계하는 것이 좋다.

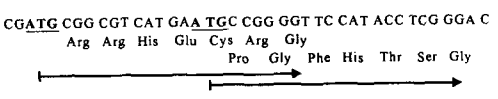

그림 2. CA의 DNA 염색체

그림 2는 DNA 염색체의 하나의 예이다. 여기에는 규칙으로 해석할 수 있는 시작 코돈이 두 개 발견된다. 이 중 첫 번째 규칙을 해석하면 다음과 같다. 우선 시작 코돈인 ATG에서 번역이 시작되고 차례로 3개

즉, 코돈에 따라 다음과 같이 번역된다.

① ATG : start codon
② Arg : order(0: i-1 → i → i+1)
③ Arg : state(4)　　④ His : A/B(A)
⑤ Glu : state(6)　　⑥ Cys : A/B(B)
⑦ Arg : operator(DIFFERENCE)
⑧ Gly : order(4: i+1 → i-1 → i)

이것을 규칙으로 표현하면 「If $\sigma_{i-1}^{j}=4$ and $\sigma_{i}^{j}=6$ then $\sigma_{i}^{j+1}=|\sigma_{i+1}^{j}-\sigma_{i-1}^{j}|$ 」 과 같이 된다. ②번의 order는 전건부 상태의 순서를 결정하는 것이고 ⑧번의 order는 후건부 상태의 순서를 결정하는 것이다. 이 규칙은 두개의 이웃 상태를 고려하고 왼쪽(i-1)과 오른쪽(i+1) 셀의 상태의 차이 값으로 다음의 상태를 구하는 규칙으로서 전건부에서 2개의 상태를 고려하므로 type 2 규칙에 해당된다. 그림 6의 두 번째 규칙도 마찬가지로 번역된다.

3.4 L-시스템의 DNA 코딩

본 절에서는 식물의 형태를 표현하는 L-시스템의 코딩에 대하여 설명한다. 문자는 줄기(F)와 잎(L)과 꽃(B)을 나타내는 3개의 문자를 사용한다. 생성규칙은 문맥 의존 및 bracket L-시스템을 사용하였으며 초기 문자열은 F로 하였다. 즉, Σ = { F, L, B}이고 α = F 이다.

표 5. L-시스템의 코돈(아미노산) 번역표

Amino Acid	개수	각도(δ)	전건부	후건부	[]의 문자수
Leu	6	30	F	F	0
Arg	6	26	L	L	1
Ser	6	35	B	B	2
Thr	4	28	F>F	[]	3
Ala	4	32	F>L	+	4
Gly	4	24	F>B	-	5
Val	4	20	F<F>F	F	0
Pro	4	40	F<F>L	L	1
Stp	3	45	-	-	-
Ile	3	50	F<F>B	B	2
Tyr	2	55	F<F	[]	3
Gln	2	60	F<L	+	4
Phe	2	65	F<B	-	5
Asp	2	70	F	F	0
Cys	2	75	L	L	1
Asn	2	80	B	B	2
Glu	2	15	F>F	[]	3
His	2	85	F>L	+	4
Lys	2	10	F>B	-	5
Trp	1	90	F<F	F	0
Met	1	5	F<L	L	1

CGATG TTC GTA TAC AGC GCT GAT TGG TAC ATA TTC CTA GTG TGA
Arg Phe Val Tyr Ser Ala Asp Trp Tyr Ile Phe Leu Val
26 F F [] 2 + F F [] 2 - F F

그림 3. L-시스템의 DNA 염색체

그림 3의 규칙을 표 5의 번역표를 이용해 번역하면 다음과 같다.

각도(δ) = 26° (CGA = Arg = 26)
F → F[+F]F[-F]F

가장 첫 번째 코돈은 식물의 가지가 벗는 각도 δ를 나타내고 그 다음은 시작코돈과 정지코돈 사이의 규칙을 번역표를 참조해 번역한다. 전건부, 후건부의 순으로 번역되며 후건부에서 []가 나오면 그 다음은 []안의 문자수를 번역한다. 만약 정지 코돈을 만나서 []의 문자가 다 채워지지 못하면 그대로 []를 닫는다.

Ⅳ. 결 론

본 논문에서는 발생/발달 모델의 하나인 셀룰라 오토마타(CA)와 L-시스템의 진화를 위한 DNA 코딩방법을 제안하였다. 발생/발달 모델은 규칙에 의해 형태를 구성해 나가는 알고리즘이기 때문에 내재적인 규칙이 포함된 복잡한 시스템의 설계에 매우 유용하다. 최근 많은 연구자들이 발생/발달 모델을 이용하여 인공두뇌(신경망)을 진화(구축)하는 연구에 뛰어 들고 있다. 기존에는 발생모델의 생성규칙을 유전자로 코드화하기 위해서 bit string을 사용하였다. 그러나 이 방법은 규칙의 수가 미리 결정되어야 한다는 결점이 있으며 규칙의 길이도 고정될 수밖에 없다. 따라서 다양한 모양의 CA 또는 L-시스템을 구축하기 힘들다. 본 논문에서 제안한 DNA 코딩방법은 규칙의 표현에 많은 융통성을 가지고 있어서 CA와 L-시스템의 규칙을 표현하는데 제약이 거의 없다. 따라서 발생/발달 모델을 진화적으로 설계하는 연구에 유용하게 활용될 수 있을 것이다.

감사의 글

본 연구는 과학기술부의 뇌과학 프로젝트(Braintec 21)의 지원으로 이루어진 결과임.

참고문헌

[1] M. Sipper, "Studying artificial life using a simple, general cellular model," *Artificial Life*, vol. 2, no. 1, pp. 1-35, 1995,
[2] A. Lindenmayer, "Mathematical models for cellular interaction in development, part I and II," *Journal of Theoretical Biology*, vol. 18, pp. 280-315, 1968.
[3] T. Yomohiro, 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. 19-30, 1996.
[4] R.A. Wallace, G.P. Sanders, R.J. Ferl, *BIOLOGY : The Science of Life 3rd eds.*, HarperCollins Publishers Inc., 1991.
[5] 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.
[6] Tstsuya Maeshiro, " Prediction of Deviant Genetic codes - why They Evolve-," *Proc. of The Fourth Intr. Symposium on Artificial Life and Robotics*, vol. 1, pp. 258-261, 1999.