

진화하는 셀룰라 오토마타 신경망의 하드웨어 구현에 관한 연구

A Study on Implementation of Evolving Cellular Automata Neural System

반창봉 · 곽상영 · 이동욱 · 심귀보
중앙대학교 전자전기공학부

Chang-Bong Ban, Sang-Young Kwak, Dong-Wook Lee and Kwee-Bo Sim
School of Electrical and Electronic Engineering, Chung-Ang University
E-mail : kbsim@cau.ac.kr

ABSTRACT

This paper is implementation of cellular automata neural network system which is a living creatures' brain using evolving hardware concept. Cellular automata neural network system is based on the development and the evolution, in other words, it is modeled on the ontogeny and phylogeny of natural living things. The proposed system develops each cell's state in neural network by CA. And it regards code of CA rule as individual of genetic algorithm, and evolved by genetic algorithm. In this paper we implement this system using evolving hardware concept Evolving hardware is reconfigurable hardware whose configuration is under the control of an evolutionary algorithm. We design genetic algorithm process for evolutionary algorithm and cells in cellular automata neural network for the construction of reconfigurable system. The effectiveness of the preposed system is verified by applying it to time-series prediction.

Keywords : Evolving Hardware, Cellular Automata, GA Processor, Neural Network, Time Series Prediction

1. 서론

ANNs와 EAs는 인공생명 연구의 대표적인 모델이다[1]. ANNs의 성능 향상을 위해 ANNs의 구조와 파라미터의 최적화를 위하여 진화적인 접근방법을 도입하고 있다. 그러나 주어진 문제가 복잡해지면 엄청난 진화시간을 필요로 하게 되고, 경우에 따라서는 진화알고리즘의 파라미터 결정에서도 부가적인 문제를 수반하는 등 문제점은 존재한다. 이러한 문제점을 근본적으로 해결하기 위해 신경망의 합성 규칙을 코드화하는 방법이 있다. 이것은 생물체의 발생 과정에서 힌트를 얻은 것으로 L-시스템 기반의 모듈형 신경망[2]과 셀룰라 오토마타 기반의 CAM-Brain[3]등이 있다. 이러한 방법은 신경망의 크기가 커지더라도 복잡성이 증가하지 않는다는 장점이 있다.

셀룰라 오토마타 신경망은(ECANS2)[4]은 발생모델의 셀룰라 오토마타와 진화 모델의 진화 알고리즘으로 이루어진다. 이 시스템은 셀룰라 오토마타에 의해 신경망의 내부 셀을 발생시키고 셀룰라 오토마타 룰을 코딩하여 유전자 알

고리즘의 개체로 간주하여 진화 알고리즘을 통해 진화를 통하여 환경에 대한 적응성을 획득한다. 그리고 여러 가지 뉴런의 배열에 따라 신경망의 기능이 결정된다. 셀룰라 오토마타 신경망이 기존의 신경망과 다른 점은 연결강도 보다는 뉴런과의 연결방식에 따라 전체 시스템의 기능이 달라진다는 점이다. 본 논문에서는 셀룰라 오토마타 신경망을 진화형 하드웨어 개념을 이용하여 FPGA에 구현하였다. 진화형 하드웨어는 유전자 알고리즘과 재구성 가능한 하드웨어의 결합체이다. 즉, 재구성 가능한 하드웨어의 구성에 필요한 bit를 진화 알고리즘의 개체로 간주하여 진화 알고리즘을 수행하여 환경에 적합한 구조로 재구성되는 시스템이다. 구현된 시스템에서는 구조적 비트를 발생시키는 룰을 코딩하여 이를 알고리즘의 개체로 간주한다. 이 시스템의 진화 알고리즘을 수행하기 위해 유전자 알고리즘 프로세서를 설계하였다. 유전자 알고리즘 프로세서 내에서는 유전자 알고리즘의 연산자인 교차와 돌연변이를 위한 연산 모듈(Operating Module)와 개체군의 재생산을 위한 재생산 모듈(Re-Production Module) 그리고,

주 제어기로 이루어져 있다. 그리고, 개체군과 적합도 값을 저장하기 위한 외부 메모리도 존재한다. 그리고, 신경망 내부의 하나의 뉴런은 진화 알고리즘 프로세서에 의해 새로 생성된 룰에 의해 비트열들을 생성하고 이 비트열들에 의해 내부구조를 스스로 바꿀 수 있도록 설계되었다. 유전자 알고리즘 프로세서를 이용하여 신경망이 자동적으로 진화함으로써, 진화형 하드웨어를 이루게 된다. 제안한 시스템의 유효성을 검증하기 위하여 시계열 예측 문제에 적용하였다.

II. ECANS2 의 개요

그림 1은 제안된 구조의 개념도이다. 셀룰라 오토마타 신경망은 발생과 진화의 두 단계를 거쳐 생성된다. 즉, 네트워크는 셀룰라 오토마타 룰[5]에 의해 생성되며, 주어진 환경에 적응할 수 있는 구조로 발전하도록 진화한다.

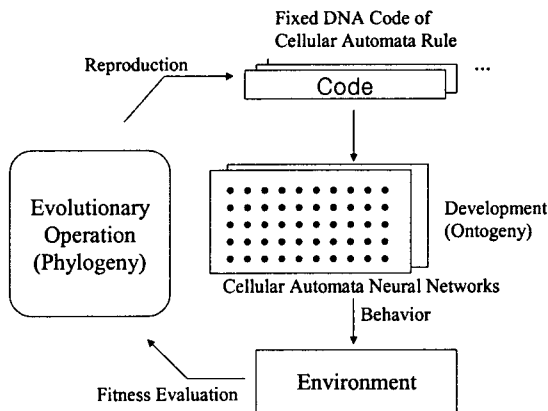


그림 1. ECANS2의 개념도

셀룰라 오토마타 신경망의 내부 뉴런은 다음 식과 같은 Nagumo-Sato[6]의 카오스 뉴런 모델을 사용한다.

$$y(t+1) = u(x(t+1)) \quad (1)$$

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

단, $y(t)$ 는 시간 t 에서 출력, $x(t)$ 는 내부 상태 값, $I(t)$ 는 입력, $u(\cdot)$ 는 단위 계단 함수, k^d 는 0에서 1 사이의 값을 가지는 감쇠계수, 상수 α 는 양의 파라미터이며 θ 는 뉴런의 문턱 값이다. 뉴런의 상태 값은 입력 값에 따라 카오틱한 행동을 보인다. 카오스 뉴런 모델에서 뉴런의 동작함수는 단위 계단 함수의 형태로 나타나기 때문에 입출력 신호는 펄스 형태이다. 따라서 신호의 세기는 펄스의 밀도 변화에 의해서 측정된다.

본 논문에서는 발생 모델로서 CA 룰을 이용

한다. 이 룰은 고정 길이의 DNA 코딩 방법을 통해 염색체가 되고, 이것이 진화 알고리즘의 개체가 된다. CA 룰이 결정되면, 초기 셀들로부터 CA 룰에 따라 발생 과정을 거쳐 신경망을 이루게 된다. CA 룰은 자신의 셀과, 좌우 셀의 연산관계를 코딩을 하게 된다. 그림 2은 발생 단계를 나타내고 있다.

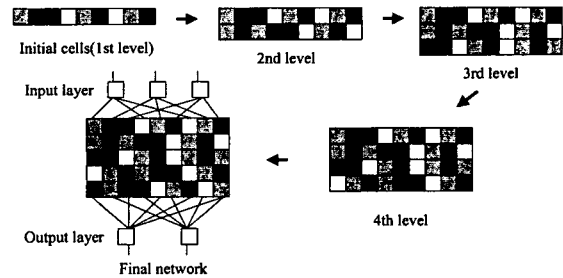


그림 2. ECANS2의 발생 단계

III. 하드웨어 설계

3.1 진화형 하드웨어

재구성 가능한 하드웨어[7]는 사용자가 소프트웨어적으로 구조를 변경할 수 있는 반도체 집적회로로서 다양한 구조의 반도체를 구현할 수 있어서 많은 분야에서 응용되고 있다. 재구성 가능한 하드웨어의 대표적인 예는 FPGA(field programmable gate array)이다. 진화 알고리즘에 의해 그 구조가 자동적으로 변하는 하드웨어를 진화 하드웨어(evolvable hardware)라고 한다. 하드웨어의 구조를 나타내는 비트 스트링을 진화 알고리즘의 염색체로 표현하여 적합도 기반으로 하드웨어의 구조를 진화하고자 하는 것이다.

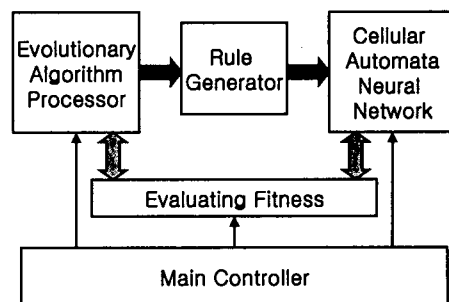


그림 3. 전체 시스템의 개요도

본 논문에서는 진화형 하드웨어를 이용하여 셀룰라 오토마타 신경망을 구현한다. 유전자 알고리즘 프로세서를 통해 유전자 알고리즘을 수행한다. 룰 발생기에서 각 개체들을 해석하여 셀룰라 오토마타 룰을 생성하고, 이 룰에 의해 신경망의 내부구조가 스스로 변화하도록 설계하였다. 그림 3은 전체 시스템의 개요도이다.

3.2 유전자 알고리즘 프로세서

유전자 알고리즘은 자연계의 생물의 진화과정을 모방한 것이다. 자연계의 생물의 진화과정은, 어떤 세대를 형성하는 개체들의 집합, 즉 개체군 중에서 환경에 대한 적합도가 높은 개체가 높은 확률로 살아 남아 재생되며, 이때 교차나 돌연변이에 의해서 다음 개체군을 형성하게 된다. 유전자 알고리즘은 구조가 간단하고 방법이 일반적이어서 응용범위가 매우 넓지만, 연산 시간이 길다는 문제점을 가지고 있다. 유전자 알고리즘의 심각한 문제인 연산 시간을 줄이기 위해서 유전자 알고리즘의 효율적인 하드웨어 구현 방법이 요구된다[8].

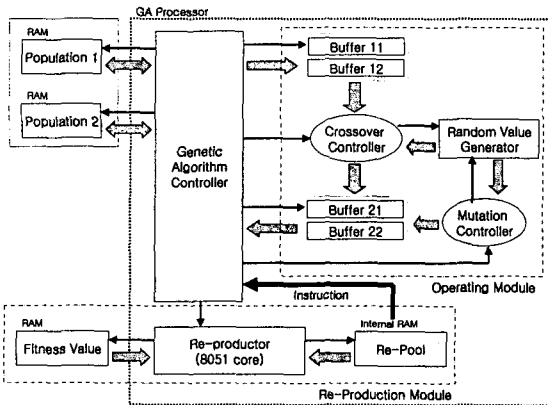


그림 4. GA 프로세서의 블록 다이어그램

그림 4는 제안된 유전자 알고리즘 프로세서의 블록 다이어그램을 나타낸다. 제안된 프로세서 내에는 유전자 알고리즘의 기본 연산자인 교차와 돌연변이 기능을 하는 연산 모듈(Operating Module)와 개체군의 재생산을 위한 재생산 모듈(Re-Production Module), 개체군을 저장하는 2개의 외부메모리, 적합도 값을 저장하는 1개의 외부메모리, 그리고 전체 기능을 제어하는 제어기(Controller)로 이루어져 있다.

기존의 유전자 알고리즘 프로세서는 하드웨어 지향 알고리즘으로, 실제 세대라는 개념을 가지지 않는 정상상태 모델을 기반으로 하였다. 그러나, 제안된 프로세서는 2개의 외부 메모리를 이용하여 세대의 개념을 갖게 된다. 그리고 각 개체에 해당하는 적합도 값을 제 3의 외부 메모리를 이용하여 저장하게 된다. 이는 새로운 세대를 구성하기 위해 재생산을 하게 될 때 적합도 값만을 참조하기 위함이다.

재생산 모듈(Re-Production Module)에서는 적합도 값을 저장하고 있는 메모리에서 적합도 값을 참조하여, 룰렛 선택을 통해 재생될 개체들을 재생 공간(Re-Pool)에 저장한다. 이때 재생될 개체들을 코드화하여 저장하고, 이 코드에 따라 유전자 알고리즘 프로세서가 동작한다. 연산 모듈(Operating Module)에서는 유전자 알고

리즘의 기본 연산인 교차와 돌연변이를 수행하게 된다. 그리고, 선택된 두 부모 개체가 교차 직후 돌연변이를 수행하여 빠른 처리 속도로 처리한다.

3.3 진화 신경망

본 논문에서는 FPGA를 이용하여 셀룰라 오토마타 신경망을 구현하였다. 룰 발생기에서 CA 룰이 생성되면, 이 룰에 따라 셀 간의 연결을 자동적으로 구성하도록 신경망 내의 셀들을 설계하여 진화 신경망을 구현하였다. 진화 신경망 내의 각각의 셀들은 셀룰라 오토마타 모듈과 연결 모듈, 그리고 뉴런 모듈의 세 가지 모듈로 구성되어 있다. 그림 5은 제안된 진화 신경망 내부 셀의 구조를 나타낸다.

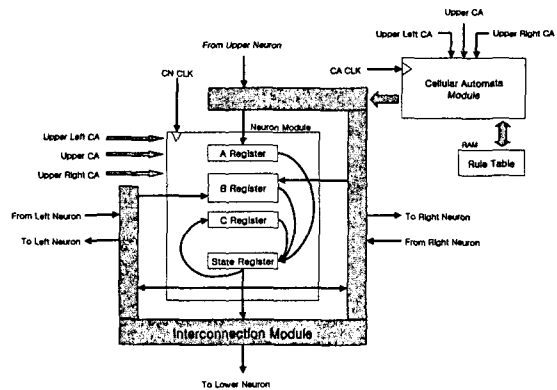


그림 5. 진화 신경망 내부의 셀의 구조

뉴런의 입력 값은 신경망 윗 층으로의 입력, 같은 층의 좌우 입력 그리고 자신으로부터의 입력의 세가로 나뉜다. 이 세가지 입력에 따라 뉴런 내부는 A, B, C Register에 의해 구현되면, 최종 결과는 State Register에 저장된다. 각 Register들은 동작 클럭이 들어 올 때마다 곱연산을 위해 저장된 값들을 쉬프트 시키고, Register에 들어오는 값을 더하게 된다. 연결 모듈에서는 셀룰러 오토마타 룰에 의해 결정된 셀의 상태를 고려하여 뉴런의 출력과, 뉴런으로 들어오는 입력을 연결하거나 연결을 차단한다.

즉, 유전자 알고리즘 프로세서에서는 셀룰라 오토마타 룰을 코딩하여 그것을 개체로 하여 유전자 알고리즘을 수행하고, 신경망 내부에서는 셀룰라 오토마타 룰을 참조하여 자동적으로 신경망의 구조를 변경하게 된다. 즉, 유전자 알고리즘 프로세서와 신경망이 결합하여 진화하는 신경망을 설계하였다.

IV. 실험 및 결과

본 논문에서는 제안된 시스템의 성능을 시험해 보기 위해 Mackey-Glass chaotic data를 이

용하여 시계열 예측 문제에 적용하였다. Mackey-Glass time series의 식은 다음과 같다.

$$\frac{dx(t)}{dt} = \frac{ax(t-\Delta)}{1+x^c(t-\Delta)} - bx(t) \quad (3)$$

문제에서 a 는 0.2, b 는 0.1, c 는 10, 그리고 Δ 는 30로 하고, 1000개의 데이터를 대상으로 한다. 그리고, 1000개의 데이터 중 전반부 500개의 데이터를 트레이닝 데이터로 하고, 나머지 500개의 데이터를 테스트 데이터로 한다. 그림 6은 컴퓨터 시뮬레이션 결과중 하나이며, 그림 7은 제안한 신경망에 의해 예측한 결과중 하나이다. 표 1은 이들 결과의 적합도 값과, MSE를 나타낸다. 이 실험에서 사용한 파라미터는 다음과 같다. 내재 차원 m 은 5, 시간 지연 τ 는 5, 네트워크의 크기는 5×10 , 네트워크의 입력은 $y(t)$, $y(t-5)$, $y(t-10)$, $y(t-15)$, $y(t-20)$, 네트워크의 출력은 $y(t+1)$, 평가함수는 $fit = e^{-10E}$ 이다.

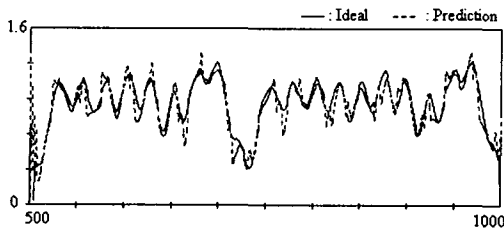


그림 6. 컴퓨터 시뮬레이션 결과

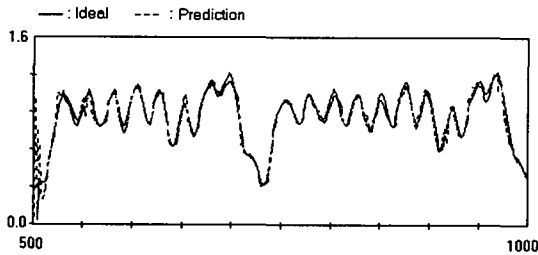


그림 7. 예측과 실제의 Mackey-Glass 데이터

표 1. 적합도와 MSE 값

	Simulation	Experiment
Fitness	0.9531	0.9677
Mean square error	0.0048	0.0052

IV. 결론

본 논문에서는 진화형 하드웨어 개념을 이용하여 셀룰라 오토마타 신경망을 구현하였다. 제안된 시스템은 유전자 알고리즘 프로세서와 재구성 가능한 셀룰라 오토마타 신경망으로 구성된다. 그리고, 셀룰라 오토마타 룰을 발생하는

룰 제너레이터를 매개로 한다. 셀룰라 오토마타 룰을 코드화하여 유전자 알고리즘의 개체로 간주하고, 유전자 알고리즘 프로세서에 의해 진화하게 된다. 그리고, 생성된 룰을 이용하여 신경망의 내부 구조를 스스로 변경하게 된다.

본 논문에서는 진화형 하드웨어의 개념을 이용하여 생물 정보 처리 시스템을 구현하였다. 즉, 자연계의 진화 및 발생의 원리를 이용한 시스템이 하드웨어 구성 될 수 있음 제시하고 그 유용성을 살펴보았다.

감사의 글 : 본 연구는 2001년도 서울시·중소기업청 산학연 공동기술개발 컨소시엄사업의 연구비지원으로 수행되었으며, 또한 과학기술부 뇌 과학 프로젝트(Braintech21)에서 연구비를 일부 지원 받았습니다. 연구비 지원에 감사 드립니다.

참고문헌

- [1] C.G. Langton ed, Artificial Life, Addison-Wesley, 1989.
- [2] 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.
- [3] Hugo de Garis, "CAM-BRAIN : The Genetic Programming 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] D.W. Lee, K.B. Sim, "Evolving Cellular Automata Neural Systems 2(ECANS2)," Proceeding of the Fourth International Symp. on Artificial Life and Robotics, coll. 1, pp. 224-227, 1999.
- [5] C.G. Langton, "Life at the Edge of Chaos," Artificial Life II, Addison-Wesley, pp. 41-91, 1992.
- [6] M. Ohta, A. Ogihara et. al., "A Study on The Mechanism of the Minimum Searching by the Chaotic Neural Network," Proceedings of International Conference on Neural Networks, pp. 1517-1520, 1995.
- [7] A. Thompson, "An Evolved Circuit, Intrinsic in Silicon, Entwined with Physics," in Lecture Notes in Computer Sciences, No. 1256, Evolvable Systems : From Biology to Hardware, Springer, 1997.
- [8] S. D. Scott, A. Samal and S. Seth, "HGA : A Hardware based genetic algorithm," Proc. ACM/SIMDA 3rd International Symposium on FPGA, pp. 53-59, 1995.