

센서-모터 제어를 위한 셀룰라 오토마타 기반 신경망 모듈의 규칙기반 결합

A Rule-based Integration of Neural Network Modules based on Cellular Automata for Sensory-Motor Controller

김경중 · 송금범 · 조성배

Kyung-Joong Kim, Geum-Beom Song and Sung-Bae Cho

연세대학교 컴퓨터과학과

Department of Computer Science, Yonsei University

요 약

자율이동로봇의 센서-모터 제어를 구축하는데 있어 로봇의 기계적인 부분과 제어기 부분을 조화시키는 것이나 외부환경과 로봇의 상호작용을 처리하는 것 등이 가장 큰 문제점이다. 이러한 문제점들을 해결하기 위해서 진화적 접근방법이 많이 사용되고 있다. 이전 연구에서는 이러한 연구선상에서 셀룰라 오토마타 기반 신경망인 CAM-Brain을 이동로봇 제어기로 진화시켰다. 그러나, 하나의 모듈로 이루어진 제어기로는 복잡한 행동을 하도록 만들기 어렵기 때문에 본 논문에서는 하위 수준의 간단한 행동을 하도록 진화된 모듈들을 결합하여 보다 상위 수준의 복잡한 행동을 하도록 하는 다중 모듈 결합방법을 제안한다. 실험결과, 간단한 행동을 하도록 진화된 CAM-Brain 모듈들을 규칙기반 방법으로 결합하여 주어진 좀더 환경에 적용할 수 있는 제어를 얻을 수 있었다.

Abstract

There are some difficulties to construct a sensory-motor controller for an autonomous mobile robot such as coordinating the mechanics and control system parts of the robot, and managing interaction with external environments. In previous research, we evolve the CAM-Brain, neural networks based on cellular automata, to control an autonomous mobile robot. In this paper, we propose the method of combining multi-modules evolved to do simple behavior in order to making more sophisticated behaviors because the controller composed of one neural network module is difficult to make complex behaviors. In experimental results, we can get the controller adapting to more complex environments by combining CAM-Brain modules evolved to do simple behavior by rule-based approach.

key words : 이동로봇 제어, 셀룰라 오토마타, 행동결합, 신경망, 행동기반 인공지능

1. 서 론

자율이동로봇의 센서-모터 제어를 구축하는데 있어 몇 가지 문제점들이 있다. 로봇의 기계적인 부분과 제어기 부분을 조화시키는 것이나 외부환경과 로봇의 상호작용을 처리하는 것 등이 가장 큰 문제점이다[1,2]. 이러한 문제점들을 해결하기 위해서 이동로봇의 제어를 진화 방식으로 구축하는 연구들이 진행중이다. 대표적인 방법으로는 신경망을 유전자 알고리즘으로 진화시키거나, 유전자 프로그래밍을 이용하거나, 퍼지 제어기와 유전자 알고리즘을 결합하는 방법이다. 이러한 진화적인 접근방법은 명확한 설계를 하지 않고도 좋은 성능을 갖는 제어를 만들어 낼 수 있는 가능성을 갖고 있다. 신경망이

다양한 방법으로 학습이 가능하고, 로봇과 외부환경과의 상호작용으로 발생하는 잡음에 강하기 때문에 신경망을 진화시키는 접근방법이 가장 유용하리라 예상된다[2].

이전 연구[3]에서는 이러한 연구선상에서 셀룰라 오토마타 기반 신경망인 CAM-Brain[4,5,6,7,8]을 이동로봇 제어기로 진화시켰다. 이 모델은 뉴런과 축색돌기 및 수상돌기로 이루어진 신경망을 셀룰라 오토마타 상에서 만들어낸다. 셀룰라 오토마타는 마치 우리의 두뇌가 수억개의 간단한 신경 세포들이 서로 연결되어 만들어져 있듯이, 간단한 규칙들로 복잡한 구조와 기능을 표현할 수 있다. 이러한 특성은 셀룰라 오토마타가 간단한 규칙들에 기반하여 복잡한 신경망을 개발할 수 있는 플랫폼(platform)이 될 수 있는 가능성을 보여준다. 게다가, 셀룰라 오토마타의 병렬성은 대규모의 신경망을 병렬하드웨어에서 빠르게 만들어 낼 수 있도록 한다[4,5,6,7,8].

그러나, 하나의 모듈로 이루어진 제어기로는 복잡한 행동을 하도록 만들기 어렵기 때문에 본 논문에서는 하위수준의 간단한 행동을 하도록 진화된 모듈들을 결합하여 보다 상위 수준의 복잡한 행동을 하도록 하는 다중

접수일자 : 2001년 6월 19일
완료일자 : 2001년 12월 12일
이 연구는 과학기술부가 지원하는 뇌과학연구 프로그램에 의하여 지원 받은 것임.

모듈 결합방법[9,10]을 제안한다. 예를들어, “장애물 피하기,” “벽이 찾기,” “집으로 돌아오기,” 등의 간단한 행동들을 결합함으로써, “벽이들을 찾아 집으로 돌아와라”라는 식의 복잡한 행동을 만들어 낼 수 있다.

본 논문은 다음과 같이 구성되어 있다. 먼저 2장에서는 본 논문에서 사용한 행동기반 이동로봇인 Khepera[5] 및 이동로봇 제어에 관한 관련연구들을 알아보고자 한다. 3장에서는 셀룰라 오토마타 기반의 진화형 신경망(CAM-Brain)에 대해 자세히 알아보고 4장에서는 다중모듈 결합방법에 대해서 자세히 살펴보고, 마지막으로 결론 및 향후 연구과제에 대해서 언급하겠다.



그림 1. 행동기반 이동로봇, Khepera (Behavior-based mobile robot, Khepera).

2. 배경

2.1. 행동기반 이동로봇: Khepera

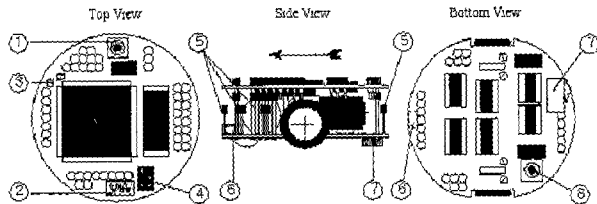


그림 2. Khepera 로봇의 각 부분의 위치 [11]. ① LEDs. ② Serial line (S) connector. ③ Reset button. ④ Jumpers for the running mode selection. ⑤ Infra-Red proximity sensors. ⑥ Battery recharge connector. ⑦ ON-OFF battery switch. ⑧ Second reset button (Parts of Khepera robot).

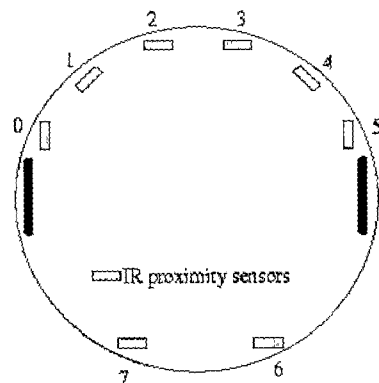


그림 3. 8개 IR 센서들의 위치(8 IR sensors of Khepera robot).

행동기반 이동로봇인 Khepera(그림 1 참조)는 로봇공학 이외에도 인공지능 및 생물학과 인지과학 등의 많은 분야의 연구를 위해 만들어졌다[1]. 이 로봇은 두 개의 바퀴를 갖고 있으며, 각각의 바퀴는 DC 모터에 연결되어 있어 독립적으로 움직일 수 있다. Khepera 로봇의 각 부분의 위치는 그림 2와 같다. 또한 8개의 적외선 센서들이 그림 3과 같은 위치에 장착되어 있다. 이러한 센서들은 적외선을 방출하고 받아들일 수 있다. 적외선을 방출하지 않고 받아들이기만 하여 빛의 강도를 측정할 수 있으며, 적외선을 방출한 후 반사된 값을 받아들임으로써 가까운 곳에 위치한 장애물의 거리를 측정할 수 있다. 이러한 측정은 주기적으로 이루어진다.

2.2. 이동로봇 제어기의 진화

이동로봇 제어기를 설계하는데 있어 크게 두 가지의 어려운 점이 있다. 첫째는 로봇의 기계적인 부분과 제어 시스템 사이의 상호작용을 미리 예측할 수 없기 때문에 두 부분을 조절하는 것이 어렵다는 것이다. 다른 하나는 이동로봇이 외부의 환경과 동적으로 상호작용하기 때문에 적절한 행동을 하도록 제어 시스템을 구축하기 어렵다는 것이다. 이것은 이전단계에서의 움직임 결과에 따라 다음단계의 입력이 결정되고[1], 외부환경과의 모든 상호작용을 모두 예측할 수 없기 때문이다[2]. 이러한 문제들을 해결하기 위한 방법으로 진화 연산과 같은 자동 절차 방법은 내부구조에 대한 자세한 사전지식이 없어도 입·출력 구조만 알면 좋은 성능을 갖는 제어기를 구축할 수 있어 상당히 유용하리라 예상된다[2]. 따라서, 이러한 진화방식을 이용한 이동로봇의 제어기 구축에 관한 연구가 최근 활발히 진행되고 있다.

신경망을 유전자 알고리즘으로 진화시키는 방법은 진화적인 접근방법 중에서도 유용한 방법이다. Cliff와 Husbands, Harvey[12,13,14]는 이러한 방향으로 깊이 있는 연구를 수행하였다. 그들은 이동로봇 제어기를 구축하는데 있어 인공적인 진화방법이 직접 설계하는 것보다 더 가능성이 있다고 주장한다. 그들은 유전자 알고리즘을 이용하여 재귀 신경망을 간단하지 않은 행동을 할 수 있는 제어기로 진화시켰다. 또한 그들은 시뮬레이션을 통해서 시각기능을 갖는 로봇의 제어기와 시각 감지 조직의 형태를 병행하여 진화시킨 결과를 제시하고 분석하였다. Floreano와 Mondada[15]은 실제 로봇의 제어기를 재귀 신경망으로 구축하였다. 그들의 목표는 적합도 평가 함수에 포함된 제한조건을 줄이고, 환경의 제공하는 것을 증가시켜 복잡한 행동을 만들어 낼 수 있다는 것을 보여주는 것이다. 이를 위해 그들은 실제 환경에서 유전자 알고리즘을 이용하여 벽과 충돌하지 않고 주어진 환경을 탐색할 수 있는 로봇의 제어기와, 간단한 적합도 평가함수를 이용하여 배터리 충전행동을 하는 제어기를 구축하였다. Walker와 Minglino[16]는 시뮬레이션 환경에서 Khepera 로봇을 효율적으로 탐험할 수 있도록 인공신경망을 진화시키는데 있어 유전자 알고리즘이 유용함을 보였다.

Koza[17]가 처음으로 제안한 유전자 프로그래밍 역시 이동로봇 제어기를 진화시키는데 많이 사용되는 방법중 하나이다. Koza와 Rice[18]는 유전자 알고리즘으로 자동적으로 생성된 컴퓨터 프로그램이 이동로봇으로 하여금 불규칙적인 모양의 방안에서 상자를 벽으로 이동시키는 작업을 수행할 수 있음을 보였다. Ebner[19,20]등은 대형

이동로봇의 간단한 이동을 위해 계층적인 구조를 갖는 제어기를 유전자 프로그래밍으로 진화시키고자 하였다. 그들은 먼저 컴퓨터 시뮬레이션을 통해 행동기반의 제어기 구조를 진화시켰으며, 그 후 실제 로봇에 적용하였다. Nordin과 Banzhaf[21,22]등은 적외선 센서를 이용하는 Khepera 로봇을 유전자 프로그래밍으로 제어 알고리즘을 진화시켰다. 그들의 제어 알고리즘은 어셈블리 프로그램이 다양한 길의 탐색체에 저장되어 있었다. 특히 그들은 실시간 온라인 학습을 위한 일반적인 방법을 보여주었다.

이 승익과 조 성배[23]는 퍼지[24] 논리 제어기를 이용하여 이동로봇을 제어하기 위해 파라미터들과, 퍼지 집합과 규칙들을 유전자 코드로 표현하고 유전자 알고리즘으로 진화시켰다. 또한, 몇 개의 기본 규칙들로부터 로봇의 행동들이 발생되었음을 보여주기 위해 진화된 제어 메커니즘을 분석하였다.

3. 셀룰라 오토마타 기반의 진화형 신경망

셀룰라 오토마타 기반의 진화형 신경망인 CAM-Brain의 기본 목표는 대규모의 뉴런으로 구성된 인공두뇌를 개발하는 것으로, 이 시스템은 셀룰라 오토마타에 기반을 두어 뉴런들의 성장과 그들간의 신호전달을 조절한다. 특히 셀룰라 오토마타의 특성상 병렬 하드웨어에서의 처리가 가능하기 때문에 엄청나게 빠른 속도로 진화시키는 것이 가능한데, 대표적인 것이 ATR 연구소와 MIT의 CBM(CAM- Brain Machine)이다.

CBM은 FPGA(Field Programmable Gate Array)에 기반한 프로그램이 가능하도록 한 논리 디바이스로, 셀룰라 오토마타에 기반한 신경망 모듈을 전자의 속도로 성장·진화시키고 수만개의 모듈로 이루어진 인공두뇌의 상태를 실시간에 변경시킬 수 있다[8]. 이러한 CBM은 수백개의 뉴런으로 이루어진 기존의 신경망에 비해 엄청나게 많은 뉴런으로 이루어진 신경망을 매우 빠른 속도로 개발할 수 있게 한다. 이러한 장점은 실제 문제를 해결하는데 도움을 줄 수 있으며 기존의 신경망 연구의 한계를 극복하는 한 방법이다.

CBM을 실제문제에 적용하기 위한 연구는 아직 부족한데, 두 개의 이진수를 입력받아 크기를 비교하는 비교기(Comparator)와, 일정한 클락(Clock)수만큼 0과 1을 내보내는 타이머(Timer), 사인(Sine)곡선을 만들어내는 등의 시뮬레이션 결과가 있다[7]. 최근에는 CBM을 인공두뇌 개발의 가능성을 보이기 위해 "Robokoneko"라고 불리는 애완용 로봇고양이를 제어하는 문제에 적용하는 연구를 수행하고있다[8].

이 신경망은 간단한 규칙들의 결합으로 복잡한 현상을 표현할 수 있는 셀룰라 오토마타의 장점을 이용한 모델로, 셀룰라 오토마타의 구성요소(상태, 주변, 규칙)에 신경망의 구조를 결정하는 탐색체를 적용시켜 하나의 신경망을 만들어낸다. 이러한 방법은 몇 개의 규칙들의 조합으로 복잡한 신경망을 만들어 낼 수 있도록 한다. 즉, 셀룰라 오토마타 공간에서 탐색체로 표현된 신경망의 구조가 유전자 알고리즘을 통해서 진화되어 최적의 구조를 찾아 낼 수 있다. 그림 4는 본 논문에서 사용한 신경망의 진화과정을 보여주고 있다.

이 모델은 셀룰라 오토마타의 상태, 주변, 규칙과 각

셀에 대응되는 탐색체의 정보에 따라 여백, 뉴런, 축색돌기, 수상돌기로 이루어진 신경망을 만들어낸다. 이때 각 셀은 그 상태에 따라 다음과 같은 역할을 한다.

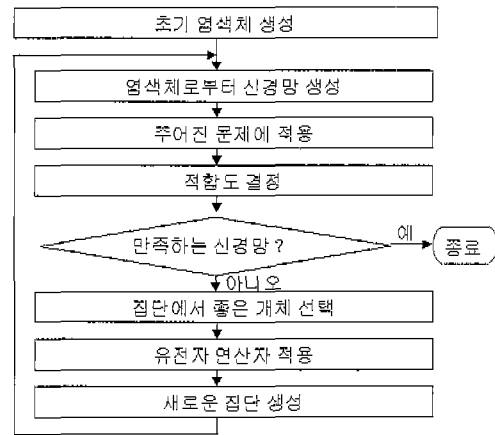


그림 4. 셀룰라 오토마타 기반 신경망의 진화과정(Evolutionary process of cellular automata based neural network).

- 여백 : 셀의 상태가 여백이면 빈 공간을 나타내며, 어떤 신호도 이 셀을 통과할 수 없다.
- 뉴런 : 이 셀은 주위의 수상돌기 셀로부터 신호를 모으고 그 값이 어떤 역치에 도달하면 주위의 축색돌기 셀로 신호를 보낸다.
- 축색돌기 : 뉴런 셀로부터 받은 신호를 주위로 전달한다.
- 수상돌기 : 주위의 셀로부터 신호를 수집하여 뉴런으로 보낸다.

3.1 신경망의 성장

이 모델은 셀룰라 오토마타 공간에 뉴런과 축색돌기 및 수상돌기로 이루어진 신경망의 구조를 만들어 낸다. 셀룰라 오토마타 공간의 탐색체에 의해 결정된 뉴런은 주위로 성장신호를 보내 주변의 여백 셀을 축색돌기나 수상돌기로 성장시킨다. 이 셀들은 다시 자신이 받은 성장신호를 주위로 보내 주변 셀들을 점점 성장시켜 하나의 신경망을 완성시킨다.

먼저 탐색체를 임의로 생성한 후 모든 셀을 여백 상태로 만든다. 그 후 주어진 확률로 셀룰라 오토마타 공간에 뉴런을 심고, 이웃한 셀에 축색돌기 및 수상돌기 성장신호를 보낸다. 성장신호를 받은 여백 셀의 상태는 성장신호의 종류에 따라 수상돌기나 축색돌기로 변하고, 상태가 변한 셀은 다시 자신이 받은 성장신호를 이웃한 셀로 보낸다. 이러한 과정이 반복되면서 모든 셀에서 상태변화가 더 이상 일어나지 않게 되어 하나의 신경망을 얻을 수 있다.

그림 5는 4 × 4 크기의 2차원 셀룰라 오토마타 공간에서 신경망이 성장하는 과정을 보여준다. 그림 5에서 빗금이 있는 셀은 여백이며, 그 안의 검은색 화살표는 탐색체에 의해서 결정된 신호전달 방향이다. 그림 5(a)는 여백으로 초기화된 공간에 뉴런을 심는 과정을 보여주고 있는데, 이 경우 뉴런의 위치는 (x2,y2)에 있는 셀이다. 그림 5(b)는 주위에 있는 여백 셀이 축색돌기와 수상돌기 상태로 변한 모습을 보여주고 있다. 그림 5(c)는 주변의 여백 셀들의 상태가 변하면서 신경망이 성장하는 모

습을 보여주고 있다.

하나의 뉴런에서 수상돌기는 신호를 모아 자신이 속한 뉴런에게 보내고, 축색돌기는 자신이 속한 뉴런에서 받은 신호를 내보낸다. 이를 위해서 신경망의 구조가 결정될 때 각 셀은 어느 방향의 주변 셀로부터 영향을 받았는지 기억하고 있어야 한다.

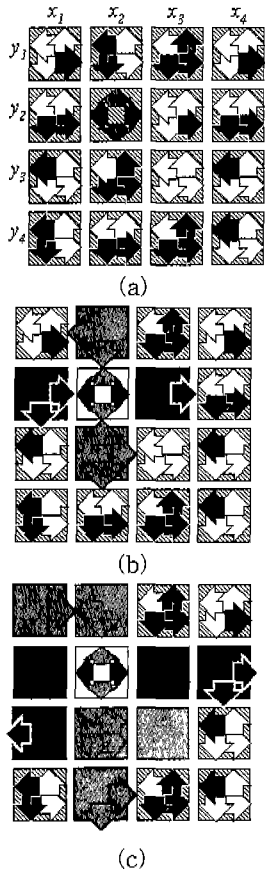


그림 5. 신경망이 성장하는 과정. (a) 뉴런 셀의 위치: (x_2, y_2), 검은색 화살표: 염색체에 의해 결정된 신호전달 방향. (b) 뉴런 셀 주변의 셀들이 축색돌기 혹은 수상돌기로 그 상태가 변함(c) 여백 셀들의 상태가 변하면서 신경망이 성장하는 모습 (Growth phase of neural network).

3.2 주어진 문제에 적용

염색체에 의해 하나의 신경망이 완성된 후 이를 문제에 적용시키기 위해서는 주어진 신경망에 신호를 입력하여 결과를 얻어내야 한다. 이때 입력과 출력은 격자형 셀룰라 오토마타 공간의 특정한 위치에 있는 셀을 통해 이루어진다. 외부로부터 신호를 받거나 내보낼 수 있는 것은 뉴런이기 때문에, 원하는 결과를 얻기 위해서는 입력과 출력으로 결정된 셀이 뉴런 상태를 가져야 한다. 그렇지만, 유전자 알고리즘을 사용하면 입력과 출력 셀의 상태를 인위적으로 뉴런으로 결정하지 않아도, 외부와의 신호전달이 잘 이루어지도록 자연스럽게 결정된다. 입력과 출력이 정해지면 외부로부터 신호를 받아들이고 내보낸다.

먼저 입력으로 정해진 셀의 상태가 뉴런이면 외부로부터 신호를 받아 계속 누적시켜, 그 값이 역치값보다

크면 흥분성 축색돌기 세포 신호를 보낸다. 이때 신호를 받은 축색돌기는 다시 이웃의 인접한 세포로 그 신호를 보낸다. 이러한 과정이 반복되어 축색돌기는 계속 인접한 세포로 신호를 보내게 되며, 이때 다른 뉴런에 종속된 수상돌기 세포들이 보낸 신호를 받으면, 이를 수집하여 자신의 뉴런이 있는 방향으로 신호를 보낸다. 이러한 과정이 반복되면서 입력 뉴런으로 보내진 신호가 여러 뉴런들을 거쳐 출력 뉴런에까지 도달한다.

그림 6은 앞에서 설명한 규칙들의 조합으로 뉴런과 축색돌기 및 수상돌기가 형성되었을 때 어떻게 신호가 전달되는지 개념적으로 보여주고 있다. 여기에서 흥분성 축색돌기로 성장한 셀에게는 흥분신호(+1)를 보내고 억제성 축색돌기로 성장한 셀에게는 억제신호(-1)를 보낸다. 수상돌기 셀은 주위의 셀에게서 신호를 수집하여 뉴런으로 보내고, 축색돌기는 뉴런으로부터 받은 신호를 주위의 셀로 보낸다.

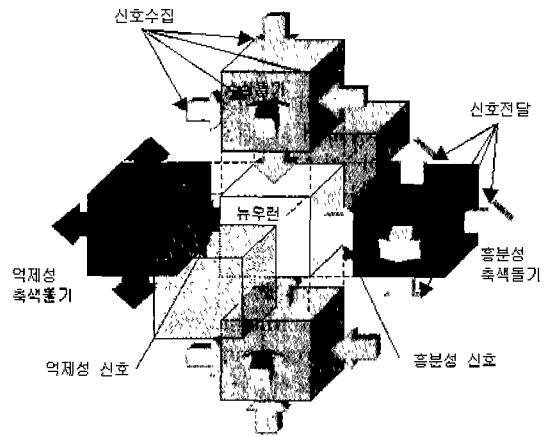


그림 6. 신호전달 과정(Signaling phase of neural network).

4. 다중모듈의 결합

이전의 연구에서는 하나의 모듈로 이루어진 제어기를 생성해내는 방법에 대해 연구하였다. 그러나 하나의 모듈로 구성된 제어기로는 복잡한 행동을 하도록 진화시키기 어렵다. 이러한 문제를 해결하기 위한 방법으로 간단한 행동을 하는 모듈들을 결합하는 방법이 있다. 즉, “직진하기,” “장애물 회피하기,” “물체 찾기,” “집으로 돌아오기,” “물체 내려놓기” 등의 간단한 행동을 하도록 만들어진 모듈들을 결합함으로써 “주어진 환경에서 물체를 찾아 집에 쌓아두어라”는 식의 상위수준의 복잡한 행동을 할 수 있는 제어기를 만들어 낼 수 있다.

이는 복잡한 기계학습 문제를 모듈형 셀룰라 오토마타 기반 신경망의 통합을 통하여 해결하는 방법으로 셀룰라 오토마타 기반의 진화형 신경망들의 결합 방법 연구의 일환인데, 각각 다른 기능능 갖도록 진화된 신경망의 결합으로 적응 행동을 하는 신경망을 만들어 낼 수 있다. 본 논문에서는 하나의 상위수준 행동을 위하여 4개의 기본행동을 정의하고, 각 행동들을 셀룰라 오토마타 기반 신경망을 이용하거나 프로그래밍을 이용하여 구축하였다. 그리고, 이러한 기본행동들을 규칙을 기반으로 결합하면 초기에 의도한 상위수준의 행동을 구현할 수 있으리라 기대된다.

4.1. 방법

본 논문에서는 다음과 같은 4개의 기본행동으로 이루어진 모듈들을 결합하여 이동로봇 제어에 적용하였다. 좀더 복잡한 행동을 위해 배터리를 추가하였는데, 이는 로봇이 움직일 때마다 배터리가 1만큼 감소하도록 시뮬레이션 하였다.

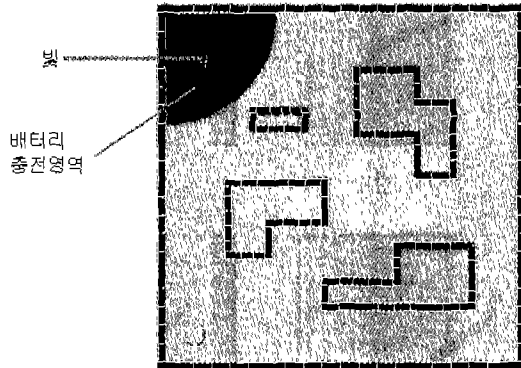


그림 7. 다중모듈 결합방법을 위한 시뮬레이션 환경 (Simulation environments).

- 배터리 충전하기: 로봇이 배터리 충전영역에 도착하면 배터리가 충전된다.
- 직진하기: 주위에 아무 것도 존재하지 않을 경우 직진한다.
- 빛 따라가기: 빛이 가장 강한 쪽으로 움직인다.
- 장애물 회피하기: 로봇의 주위에 장애물이 있을 때 장애물을 피한다.

이러한 기본 행동들은 그림 7과 같은 환경에서 계속 움직여나가기 위해 필요하다. 즉, 주어진 환경에서 살아남기 위해서는 배터리가 방전되기 전에 배터리 충전 영역에 도착하여 배터리를 충전해야 한다. 이러한 기본행동들은 모두 앞의 실험과 마찬가지로 Khepera 시뮬레이터에서 수행되었으며, 기본적인 환경변수 역시 유효하다.

모듈들의 결합을 통해 상위 수준의 행동을 만들어내는 것의 타당성은 생물학의 연구결과에서 엿볼 수 있다 [10]. 본 논문에서는 IF-THEN 규칙을 이용하여 기본행동들을 결합하였다. 이 방법은 현재 로봇의 센서값을 통해 판단한 상황에 따라 필요한 모듈을 작동시켜 적절한 행동을 하도록 한다. 본 연구에서 사용한 규칙은 다음과 같다.

```

IF (배터리 충전 영역)
    배터리 충전하기 모듈 실행
ELSE IF (배터리 < α) AND (빛 감지 센서의 최소값 ≤ γ)
    IF (거리 감지 센서의 최대값 ≤ β1)
        빛 따라가기 모듈 실행
    ELSE
        장애물 회피하기 모듈 실행
ELSE IF (거리 감지 센서의 최대값 ≤ β2)
    직진하기 모듈 실행
ELSE
    장애물 회피하기 모듈 실행
    
```

이와 같은 규칙은 매우 간단하며 명확하여 각 모듈의 동작과정을 잘 살펴볼 수 있다. 이러한 규칙이 실제로 적용되기 위해서는 다음과 같은 상수들이 결정되어야 한다.

- α: 배터리 감지 센서의 값이 α보다 작으면 배터리 충전필요.
- β₁, β₂: 거리감지 센서의 값이 β₁, β₂보다 크면 장애물로 인지.
- γ: 빛 감지 센서의 값이 γ보다 작으면 빛으로 인지.

이러한 상수들의 값에 따라 로봇의 행동이 매우 달라진다. 가령 α값이 커지면 배터리 충전영역의 주변에서만 맴돌고, 그 값이 작아지면 배터리 충전영역에 도달하기 전에 배터리를 다 소진할 것이다. 또한, β 값이 커지면 로봇이 벽과 충돌하는 회수가 늘어나며, γ 값이 너무 작으면 배터리 충전영역에서 멀리 떨어진 곳에서는 그 위치를 알 수 없을 것이다.

4.2. 실험결과 및 분석

본 실험은 Khepera 시뮬레이터를 수정하여 사용하였으며, 4개의 기본행동을 수행하는 모듈은 2종류가 있다. 배터리 충전하기와 직진하기 모듈은 프로그램을 통해 만들어졌고 나머지 빛 따라가기와 장애물 회피하기 행동은 CAM-Brain 모듈을 진화시켜 구현하였다.

배터리 충전하기 행동은 신경망을 통해서 만들어지지 않았는데, 이는 이 행동 자체가 매우 간단하기 때문이다. 기본 알고리즘은 배터리 충전영역 (그림 17 참조)에 로봇이 도달하면 배터리가 충전되고 그 이외의 지역에서는 배터리가 충전되지 않는다는 것이다. 배터리 충전영역인가 아닌가의 판별은 여러 가지가 있을 수 있지만 여기에서는 거리를 갖고 측정하였다. 그러나, 실제 환경에서는 로봇의 위치를 좌표로 표시하기 어렵기 때문에 다른 방법을 사용해야 하는데 바닥을 감지할 수 있는 센서를 장착함으로써 해결할 수 있을 것이다 [15].

주위에 아무런 장애물이 없을 때 하는 행동으로 부조건 직진을 하도록 만들어졌다. 이 행동 자체는 매우 간단하기 때문에 신경망을 진화시켜 만들지 않고 단순히 로봇 바퀴의 속도를 직접 조정하여 만들어 내었다. 그림 8은 완성된 방법을 이용하여 만든 제어를 로봇에 적용했을 때의 실제 경로를 보여주고 있다.

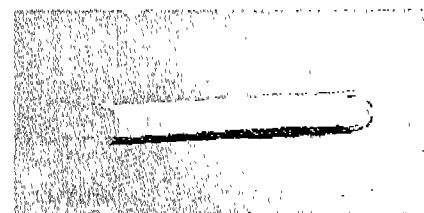


그림 8. 아무 것도 없을 때 직진하는 행동 (Going straight behavior module).

빛 따라가기 행동은 빛이 가장 강한 방향으로 움직이도록 제어를 진화시켰다. 빛은 배터리 충전영역이 있는 것을 알려주기 때문에 빛을 따라가는 행동은 매우 중요한 행동이다. 이를 위해 Khepera 시뮬레이터의 빛 감지 센서의 값을 입력으로 사용했으며, 적합도는 매우 간

단한 방법으로 주었다.

$$\text{적합도} = S * (1 - \sqrt{V}) * (c * \text{목표지점과의거리}) \quad (5)$$

S : 두 바퀴 속력의 평균

V : 두 바퀴 속도의 차

c : 1 (벽과의 충돌이 없을때)

1/2 (벽과의 충돌이 있을때)

이러한 적합도는 직진과 빠른 속력 그리고 빛과의 거리가 가까워질수록 그 값이 커진다. 따라서 진화된 제어기는 빛이 강한지점으로 빨리 움직일 수 있을 것이다. 하나의 개체당 로봇의 방향을 달리하면서 4번을 실험하고 그 평균을 적합도로 정하였다. 적합도의 평가가 일반적으로 이루어지기 위해서는 보다 다양한 환경과 많은 반복 실험을 필요로 하지만 그에 따라 평가시간이 상승하기 때문에 어려움이 따른다. 현재의 연구에서는 4번의 반복 실험을 통해 적합도 평가의 정확성을 높였다.

그림 9는 각 세대에 따른 최대 적합도의 변화를 그림 10은 적합도의 평균을 보여준다. 그림 11은 완성된 제어기의 경로를 보여주고 있다. 그림 11(a)는 로봇의 방향이 0도일 때 (b)는 90도 일 때 (c)는 180도 일 때 (d)는 270도 일 때 움직이는 로봇의 경로를 보여주고 있다.

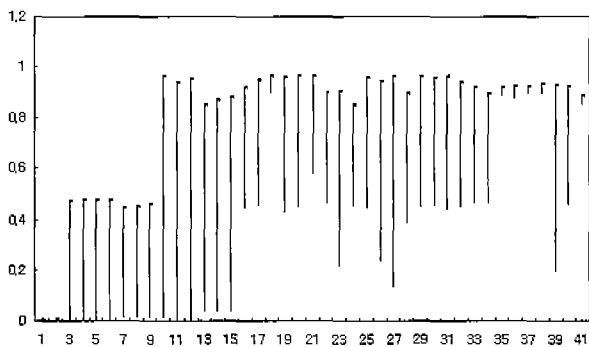


그림 9. 세대에 따른 최대 적합도 변화(Best fitness in each generation).

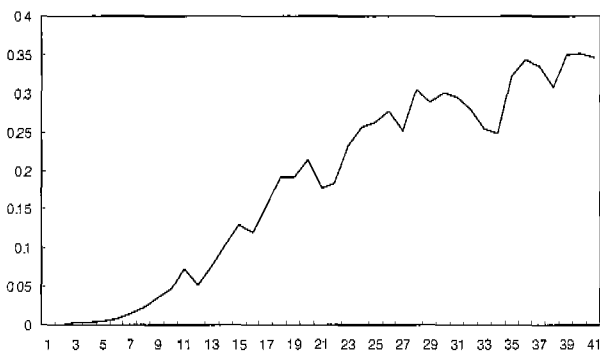


그림 10. 세대에 따른 평균 적합도의 변화(Average fitness in each generation).

장애물이 나타났을 때 그 장애물을 피하는 행동은 매우 어렵고 복잡한 행동이다. 이러한 행동을 만들어 내기 위해서 집중적 진화의 방법을 사용하였다. 이러한 방법을 통하면 좀더 효율적이고, 일반적인 제어기를 만들 수

있다. 이는 앞의 집중적 진화에서 그 가능성을 보여 주었는데 장애물 회피하기 행동을 만들기 위하여 이 결과를 이용하였다. 그림 12는 진화된 제어기가 어떻게 다양한 형태의 장애물을 피하는 것을 보여준다.

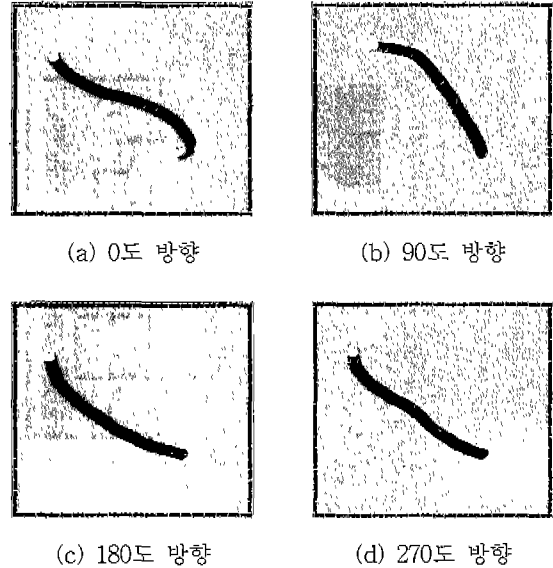


그림 11. 성공한 로봇의 경로(Trajectories of successful robot).

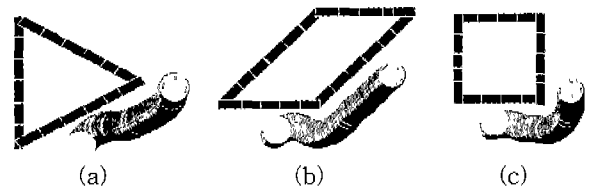


그림 12. 다양한 형태의 장애물 피하기(Avoiding obstacle behavior module).

결합은 IF-THEN 규칙을 이용하였다. 위에서 설명한 알고리즘을 이용하여 4개의 기본 행동들을 결합함으로써 좀더 복잡한 행동을 만들어 낼 수 있다. 우선 로봇으로부터 센서값을 받아 어떤 모듈을 선택할 것인가를 앞 절의 알고리즘에 의해 결정한다. 선택된 기본행동이 실행된 후 다시 센서값을 받아 어떤 행동이 선택될 것인가를 결정한다. 이러한 과정이 반복되면서 로봇이 움직인다.

그림 13은 주어진 환경에서 로봇이 움직이는 경로를 보여주고 있다. 시각 위치에 따라 로봇이 움직이는 경로가 다르고, 능력 역시 다르다. 배터리는 로봇이 한번 움직일 때마다 1씩 줄고, 한번 충전하면 배터리는 2500이 된다. 따라서 약 2500번 로봇이 움직일 수 있다. α 는 최대 배터리 양의 2/3, β_1 은 200, β_2 는 250으로 정하였다. 그림 13의 (a), (b), (c)는 로봇의 위치를 여러 가지로 변화시켰을 때 로봇의 경로를 보여주고 있다. 그림 13(a)는 약 5000번의 단계, 그림 13(b)는 약 10000번의 단계, 그림 13(c)는 14000번의 단계를 움직였다. 그림 14는 로봇이 16798번 움직였을때의 행동선택순서를 보여준다. 표 1은 로봇이 16798번 움직이는 동안 각 행동의 선택 회수를 보여준다.

5. 결론 및 향후 연구과제

본 논문에서는 이전연구의 확장을 위해 최종적으로 인공두뇌의 개발을 목표로 하는 셀룰라 오토마타 기반의 진화형 신경망의 다중모듈 결합방법을 제안하였다. 이전 연구에서는 간단한 환경에서 진화된 이동로봇의 제어가 주어진 문제를 잘 해결할 수 있음을 보였고, 특히 진화된 제어를 분석하여 뉴런들과 그 연결이 어떻게 이루어져 있으며, 어떠한 역할을 하는지 밝혀내고자 하였다. 그러나, 실험한 환경이 간단하기 때문에 이 신경망의 문제 해결 능력이라든지, 진화 가능성을 가늠하기에는 부족하다. 따라서, 본 논문에서는 복잡한 문제를 해결하기 위해 셀룰라 오토마타 기반의 신경망 모듈들의 결합방법을 제안하였다.

다중모듈 결합방법은 간단한 행동들을 하도록 진화되거나 프로그램된 모듈들을 결합하여 복잡한 환경에서의 이동로봇 제어에 적용하여 보았다. 실제로 복잡한 환경에 적용하는 이동로봇 제어를 만들어내는 것은 어려운 문제이다. 따라서, 본 연구에서는 간단한 행동을 하도록 만들어진 여러개의 모듈들을 IF-THEN 규칙을 이용하여 결합하여 그 결과를 보여주었다. 이 방법은 간단하지만 상수 값들을 조정하면서 좀더 좋은 규칙을 찾을 수 있었고, 아직 미흡하지만 어느 정도 만족할 만한 결과를 얻을 수 있었다. 결과적으로 본 논문은 진화된 셀룰라 오토마타 기반 신경망 모듈들의 결합 가능성을 보여주었다.

그러나, 인공두뇌의 개발이라는 입장에서 셀룰라 오토마타 기반의 진화형 신경망에 대한 더 많은 연구가 필요하다. 본 논문에서는 셀룰라 오토마타 기반의 신경망 모듈들의 결합 가능성을 보여주었지만, 대규모의 모듈들을 결합할 수 있는 방법을 찾아야 하고, 그 가능성을 검사해 보아야 할 것이다. 또 다른 문제는 본 논문의 연구는 제어의 문제에 국한되어 있으나, 실제 생물학적 두뇌의 또 다른 중요한 기능인 인식문제에 대한 연구도 필요하다.

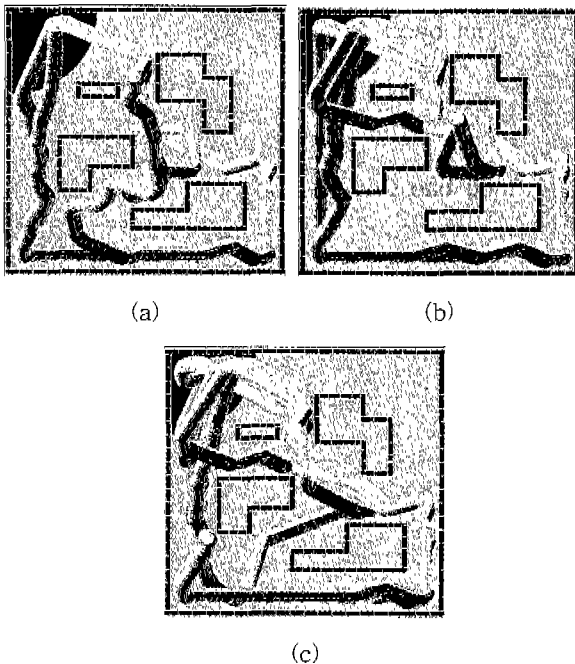


그림 13. 다중모듈 결합방법의 시뮬레이션 결과(Simulation results of rule-based integration).

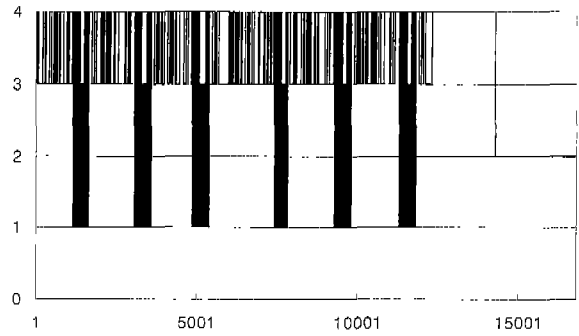


그림 14. 행동선택 순서 (1=배터리 충전하기, 2=빛 따라가기, 3=장애물 피하기, 4=직진하기) (Action sequence of robot).

표 2. 각 행동별 선택 회수 (Selected number of each behavior module).

배터리 충전하기	빛 따라가기	장애물 피하기	직진하기	합계
1478	2500	4370	8450	16798

참고 문헌

- [1] I. Harvey, P. Husbands and D. Cliff, "Issues in evolutionary robotics," Proc. of the Second Int. Conf. on Simulation of Adaptive Behavior (SAB92), pp. 364-373, Cambridge MA, 1993.
- [2] S. Nolfi, D. Floreano, O. Miglino and F. Mondada, "How to evolve autonomous robots: different approaches in evolutionary robotics," Proc. of the Int. Conf. on Artificial Life IV. pp. 190-197, 1994.
- [3] 송금범, 조성배, "셀룰라 오토마타 기반 신경망의 진화 및 행동분석," 한국정보과학회 논문지, 제 26권, 제 4호, pp. 453-461, 1999.
- [4] H. de Garis, "CAM-Brain: ATR's billion neuron artificial brain project," Proc. of Int. Conf. on Evolutionary Computation, pp. 886-891, Nagoya, Japan, 1996.
- [5] F. Gers, H. de Garis and M. Korkin, "CoDi-1Bit: A simplified cellular automata based neuron model," Proc. of Artificial Evolution Conf., Nimes, France, 1997.
- [6] M. Korkin, H. de Garis, F. Gers and H. Hemmi, "CBM(CAM-BRAIN MACHINE): A hardware tool which evolves a neural net module in a fraction of a second and runs a million neuron artificial brain in real time," Proc. of Genetic Programming Conf., Stanford, USA, 1997.
- [7] N.E. Nawa, H. de Garis, F. Gers and M. Korkin, "ATR's CAM-Brain Machine (CBM) simulation results and representation issues," Proc. of Genetic Programming Conf., USA, 1998.
- [8] H. de Garis and 조성배, "CAM-Brain:ATR의 인공두뇌 프로젝트," 정보과학회지, 제17권, 제5호, pp.

20-26, 1999.

[9] W. Banzhaf, P. Nordin and M. Olmer, "Generating adaptive behavior using function regression within genetic programming and a real robot," Int. Conf. on Genetic Programming, pp. 35-43, 1997.

[10] M.J. Mataric, "Designing and understanding adaptive group behavior," Adaptive Behavior, vol. 4, no. 1, pp. 51-80, 1995.

[11] K-Team, Khepera Simulator Version 5.02 User Manual, 1999.

[12] D.T. Cliff, P. Husbands and I. Harvey, "Analysis of evolved sensory-motor controllers," Proc. of the 2nd European Conference on Artificial Life (ECAL93), pp. 192-204, 1993.

[13] D.T. Cliff, I. Harvey and P. Husbands, "Incremental evolution of neural network architectures for adaptive behaviour," Proc. of the First European Symposium on Artificial Neural Networks, pp. 39-44, Brussels, Belgium, 1993.

[14] I. Harvey, P. Husbands and D. Cliff, "Seeing the light: Artificial evolution, real vision," Proc. of 3rd Int. Conf. on Simulation of Adaptive Behavior '94, pp. 392-401, 1994.

[15] D. Floreano and F. Mondada, "Evolution of homing navigation in a real mobile robot," IEEE Trans. Systems, Man, and Cybernetics, vol. 26, no. 3, pp. 396-407, 1996.

[16] R. Walker and O. Miglino, "Simulating exploratory behavior in evolving artificial neural networks," Proc. of the Genetic and Evolutionary Computation Conf., pp. 1422-1428, Orlando, USA, 1999.

[17] J.R. Koza, Genetic Programming: On The Programming of Computers by Means of Natural Selection, MIT Press, Cambridge, Massachusetts, 1992.

[18] J.R. Koza, and J.P. Rice, "Automatic programming of robots using genetic programming," Proc. of the 10th National Conf. on Artificial Intelligence, pp. 194-201, 1992.

[19] M. Ebner, "Evolution of a control architecture for a mobile robot," Proc. of the 2nd Int. Conf. on Evolvable Systems: From Biology to Hardware, pp. 303-310, Lausanne, Switzerland, 1998.

[20] M. Ebner and A. Zell, "Evolving a behavior-based control architecture - From simulations to the real world," Proc. of the Genetic and Evolutionary Computation Conf., pp. 1009-1014, Orlando, USA, 1999.

[21] P. Nordin and W. Banzhaf, "Real time control of a Khepera robot using genetic programming," Control and Cybernetics, Vol. 26, No. 3, pp. 533-561, 1997.

[22] P. Nordin, W. Banzhaf and M. Brameier, "Evolution of a world model for a miniature Robot using genetic programming," Robotics and Autonomous Systems, Vol. 25, pp. 105-116, 1998.

[23] S.B. Cho and S.I. Lee, "Mobile robot learning by evolution of fuzzy controller," Journal of Intelligent and Fuzzy Systems, Vol. 6, pp. 91-97, 1997.

[24] L.A. Zadeh, "Fuzzy sets," Information and Control, Vol. 8, pp. 338-353, 1965.

저 자 소 개



김경중 (Kyung-joong Kim)

2000년 2월: 연세대학교 컴퓨터과학과 졸업
(학사)

2000년 3월 ~ 현재: 연세대학교 컴퓨터과학과
석사과정 재학중

관심분야: 인공지능, 진화연산, 검색엔진, 이
동로봇제어

송금범 (Guem-beom Song)

2001년 2월: 연세대학교 컴퓨터과학과 졸업(석사)

2001년 3월 ~ 현재: (주)베리텍 근무

조성배 (Sung-bae Cho)

제 11권 3호(pp. 223~230) 참조