

A-life(Artificial Life) 기반의 생태계 시뮬레이션 시스템 개발

한국기계연구원
박종민* · 박진형

Development of A-life(Artificial Life) based Ecosystem Simulation System

Korea Institute of Machinery and Materials

Jong-min Park · Jin-Hyoung Park

요 약

최근 환경의 중요성이 강조되면서 대규모 개발 사업 또는 대형 해상사고 등이 환경과 생태계에 미치는 영향을 예측하고자 하는 요구가 많이 발생하고 있다. 우리의 환경과 거기에 살고 있는 생물들은 비선형으로 행동하고 생존하므로 환경의 변화에 따른 생태계의 변화를 예측하기 위한 시뮬레이션 시스템은 비선형계를 표현해야 한다. 본 논문에서는 미션형성을 충실히 반영하고 있는 A-life와 그것을 기반으로 해서 만들어진 생태계 시뮬레이션 시스템의 개발에 관해 기술하고 있다.

1. 서론

최근 환경문제가 여러 가지 측면에서 많이 논의되고 있고 그 중요성도 크게 강조되고 있다. 예를 들어 선박사고로 인한 해양 오염, 새로운 항만건설에 의한 환경파괴, 간척사업에 의한 환경파괴 등에 대한 관심이 크게 높아지고 있는 상황이다. 그러나 이들 오염 및 환경파괴가 생태계에 미치는 영향의 예측 및 정량화에 관한 거시적인 관점에서의 체계적인 연구는 미진한 상태이다. 이에 대한 해법으로 본 논문에서는 A-life를 이용한 생태계 시뮬레이션 시스템을 제안하고 있다.

A-life는 기존의 생명체 또는 새로운 생명체의 형질과 존재 방식을 여러 가지 형태로 모델링하여 생명체를 좀더 정확하게 이해함으로써 그 생명체가 만들어낼 수 있는 여러 가지 변화를 예측할 수 있게 해준다 [1].

본 논문에서는 2장에서 A-life의 일반에 대해 서술하고 3장에서 A-life 기반의 시뮬레이션 시스템의 개발에 대하여 기술하였다. 4장에서는 개발된 시뮬레이션 시스템이 생태계의 특성을 얼마나 잘 나타내고 있는지에 대한 실험 및 결과 분석을, 마지막 5장에서는 결론 및 향후 과제를 제시하고 있다.

2. A-life(Artificial Life)

2.1 A-life의 개념들

A-life는 컴퓨터 과학, 전자공학, 생물학, 물리학, 화학,

철학 등의 학문과 인간이 만든 여러 가지 미디어를 사용하여 생명체가 지닌 적응, 진화, 자기복제 등의 특성을 연구하여 이를 통한 생명 현상 자체의 이해 및 나아가 새로운 개념의 생명체의 창조에까지 목표를 두는 학문이다 [2].

A-life는 본질적으로 생명에 관한 화두를 내포함으로써 생명의 정의에 대해 고민하지 않을 수 없으나 아직 그 누구도 생명이란 무엇인가에 대한 질문에 명확하게 답할 순 없다. 따라서 생명 현상의 객관적이고 기본적인 특징들을 다음과 같이 나열하여 이 물음에 근접할 수 있게 한다.

- 스스로의 또는 관련된 유기체와의 자기 복제
- 물질과 에너지 변환을 위한 신진대사(metabolism)
- 환경과의 기능적 상호 행동
- 부분과의 상호 의존(Interdependence)
- 환경의 혼란(perturbation)하에서의 안정성
- 진화 능력
- 성장 또는 팽창(expansion)

2.2 A-life의 방법

생명 개념 중 연속체 개념에서는 복잡계 또한 한 형태의 실아있는 유기체로 볼 수 있으며 이러한 복잡계는 우리가 살고 있는 사회, 경제 등도 포함되며 이들은 계의 무질서적인 측면에서는 케어아스(chaos) 범주에 속하며 페턴의 형태로는 프랙탈(fractal)로 표현된다. 바로 이러한 복잡계의 특성이 생명현상의 환경이 될 수 있으므로 케어아스 영

역으로부터 생명현상으로 접근하는 방법인 비선형계의 칭발성 및 자기 유지성을 연구하는 진화 역학(Evolutionary Dynamics)의 방법이 있다 또한 자연 생명체의 DNA 및 이를 통한 자기 복제 및 진화를 묘사하는 한 방법인 진화적 방법(Evolutionary Approach)이 있다 여기에는 기본적으로 유전 알고리즘 또는 진화 알고리즘을 사용하며 L-System, 모의 담금질(Simulated Annealing), 신경망 등의 방법을 사용하며 기본적으로는 폰 노이만의 세포 자동자(cellular automata) 이론과 튜링 머신(turing machine) 이론을 근간으로 한다 지금까지의 내분부의 A-life의 적용 및 산출 형태는 컴퓨터에 기반한 인실리코(In-silico) 생명체이며 이들은 다음과 같은 생물학에서 따온 개념과 방법들을 디지털 컴퓨터화 된 개념으로 사용한다.

- 게놈(Genome)과 페놈(Phenome)
- Reproduction
- Mating
- Mutation, Transposition, Zapping

2.3 A-life의 유형 및 응용

A-life의 현 상태는 초기 인공 지능의 역사와 비슷한 점이 있다 그러나 인공지능의 한계진과 접근방식의 오류를 극복하는 방법으로 A-life는 현재 다음과 같은 방법 또는 과정의 형태로 구분할 수 있다.

- Evolutionary Computer Processes
- Biomorphs and Ontogenetically Realistic Processes
- Robotics
- Autocatalytic Networks
- Cellular Automata
- Artificial Nucleotides
- Cultural Evolution

3. A-life 생태계 시뮬레이션 시스템 모델링

A-life 시뮬레이션 시스템은 시뮬레이션 모델로 functional model을 이용하고 실행 모델로는 시분할 실행(time-slicing execution)으로 구현되었다 위의 모델들은 시뮬레이션에서 병렬성을 잘 반영할 수 있으므로 A-life 개체의 시뮬레이션에 적합한 모델이다 표 1은 시분할 실행 알고리즘을 나타내고 있다 본 시스템은 환경요소들을 나타내기 위한 CA(Cellular Automata)부분과 A-life 개체를 나타내는 부분으로 나누어 구현하였다. 또한 생태계의 복잡계 시뮬레이션이라는 관점에서 보면 실제의 생태계에서는 개체의 수가 너무 많기 때문에 모델링에서의 CA cell을 모델링을 원하는 공간으로 표현하고 하나의 개체는 실제 개체단위를 의미하기보다는 이를 점유하는 해당 종의 누리 단위로 취급한다

```

Algorithm Template for Time-Slice Execution
Begin Main
    Initialize state variables
    Loop through all functions
    While not end of simulation
        Update functions
    End While
End Main
Procedure UpdateFunctions
    For all functions
        Switch on function type
            Case type 1 apply function 1
            Case type 2 apply function 2
            ...
            Case type n apply function n
        End For
    End For
    Increment time using time slice
    
```

표 1 시분할 실행 알고리즘

3.1 CA(Cellular Automata) 환경

CA는 인접한 cell들과 상호작용하는 동일하게 프로그램된 cell들의 컴퓨터 배열이다 즉, 각 cell마다 유한상태 오토마타를 갖춘 D-차원의 격자이다 [3] 본 시스템에서는 생태계의 환경은 표현하기 위해 2차원 CA를 사용하였다 각 cell들은 생태계의 환경적인 특성인 영양분, 온도, 빛의 양, 그 cell을 점유하는 개체의 식별자 등을 속성으로 가지며 국지적 상호작용(local interaction), 법칙의 동일성(homogeneity of law), 병렬 계산(parallel computation) 등의 CA의 기본 요건을 만족시키도록 구현하였다.

3.2 A-life 개체들의 동기화(synchronization)

병렬성을 지닌 시뮬레이션은 시뮬레이션 각체들의 동기화가 중요하다 즉, 같은 시간간에서 서로 독자적으로 움직이다기 서로에게 여러 가지 형태로 영향을 주기 때문에 실제 같은 시간간을 공유하기 위해서는 공간뿐만 아니라 시간의 공유도 중요하며 이를 위해서 동기화는 반드시 이루어져야 한다 Functional model을 이용한 병렬 시뮬레이션의 동기화는 shared clock 또는 shared FEL(Future Event List)로 구현할 수 있다 [1] Shred FEL 방법은 shared clock을 개선한 알고리즘이지만 A-life개체들을 병렬로 실행시킬 경우 future event list를 구성하기가 복잡하고 conflict를 일으킬 요소가 많은 이유로 shared clock 방법을 기반으로 해서 쓰레드를 이용한 동기화 알고리즘을 사용했다. shared clock에 thread의 이용은 다음과 같은 것들을 효율적이고 정확하게 처리하기 위해서 이루어졌다.

- A-life 개체는 언제든지 태어날 수 있고 또 죽을 수 있어야 한다
- 동시성 제어(concurrency control)를 통해 병렬성을 구

```

Thread PulseTime // Generates Time Pulse
While (Simulation On)
    Generate UpdatePulse;
    Wait until UpdatePulse propagates to all Thread.

Thread AliveThread // decide and act
FirstWait = TRUE,
While (Alive)
    if (FirstWait) ,
    else FirstWait = FALSE,
        Wait for UpdatePulse;
    Do Action,
    Send UpdatePulse ReceivedSignal to Thread PulseTime,

Thread CATHread // CA Update
FirstWait = TRUE,
While (Simulation On)
    if (FirstWait) ;
    else FirstWait = FALSE;
        Wait for UpdatePulse,
    Update All CA cells,
    Send UpdatePulse ReceivedSignal to Thread PulseTime,
    
```

표 2 쓰레드를 이용한 동기화 알고리즘

현할 수 있어야 한다

표 2는 쓰레드를 이용한 shard clock 동기화 알고리즘을 나타내고 있다.

3.3 A-life 개체의 구현

A-life 개체는 상태를 나타내는 부분과 행동을 나타내는 두 가지 부분으로 구현하였다. A-life 개체의 상태를 나타내는 요소들로는 체중, 생체 에너지, 나이, 종(species) 특성, 현재 좌표 등으로 구성하였다. 행동의 정의와 구현은 쓰레드를 기반으로 하였다. 즉, A-life 개체의 탄생과 동시에 쓰레드가 하나 생성되며 사멸시 그 쓰레드는 소멸되는 데 해당 A-life 개체가 살아있는 동안의 모든 행동과 의사결정은 이 쓰레드가 CA정보를 획득하여 수행되며 행동의 결과에 의한 개체의 상태변화는 개체의 속성값의 변화로 반영된다

4. A-life 생태계 구현 예

구현 예로 초원에서의 초식동물이 개체수를 늘려가는 과정과 군집생활을 하는 모습을 모델링하였다. Windows NT 운영체제에서 Visual C++와 MFC를 사용하였고 A-life 개체 및 CA는 각각 객체 지향적으로 모델링하였으며 모든 개체별 행동은 독립적이며 개체의 진화와 개체으로부터의

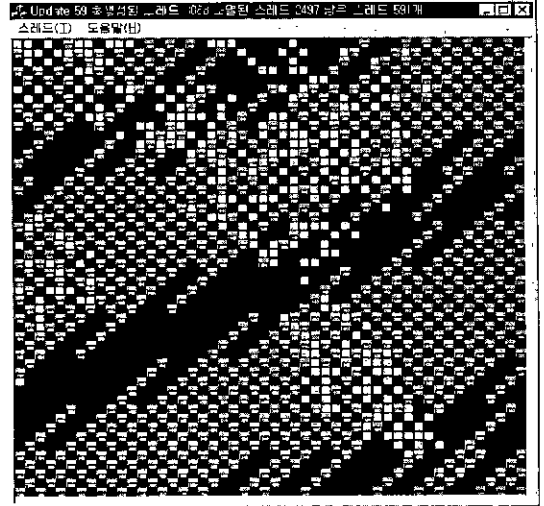


그림 1 CA환경과 1차 소비자 시뮬레이션

폐능 변환 과정은 구현되지 않았다. 따라서 개체들의 초기 폐능은 사용자에게 의해서 결정된다. 또한 CA 환경은 실제와 유사하도록 리듬을 가지고 변화하며 이것은 초원이 계절을 가지는 것과 같은 효과를 발휘한다.

5. 결론

본 연구에서는 A-life를 기반으로 생태계와 같은 비선형 복잡계를 모델링하고 시뮬레이션 하는 방법을 제안하였으며 프로토타입 시스템을 만들어 그 유효성을 검증하였다. 향후 연구에서는 다양한 종과 진화와 변종을 통한 좀 더 생명체에 가까운 행동 양식을 가진 A-life 개체 및 생태계를 모델링하여 실제 환경에서의 영향 평가 및 자원 변화에 따른 생태계 분석, 예측을 위한 도구로 사용될 수 있을 것이라고 생각된다.

참고 문헌

- [1] Rudy Rucker, *Artificial Life Lab*, Waste Group Press, 1993
- [2] Ellen Thro, *Artificial Life Explorer's Kit*, Sams Publishing, 1993
- [3] Scott Robert Ladd, *C++ Simulations and Cellular Automata*. M&T Books, 1995
- [4] Paul A. Fishwick, *Simulation Model Design and Execution*, Prentice Hall, 1995