강화학습을 이용한 멀티 에이전트 시스템의 자동 협력 조정 모델*

(An Automatic Cooperative Coordination Model for the Multiagent System using Reinforcement Learning)

정 보 윤** 윤 소 정*** 오 경 환****
(Bo-Yoon Jung) (So-Jeong Youn) (Kyung-Whan Oh)

요 약 최근 에이전트 기반 시스템 기술은 소프트웨어 시스템의 개념화, 설계, 구현을 위한 새로운 패러다임을 제공하며 많은 기대를 받아왔다. 특히 멀티 에이전트 시스템은 분산적이고 개방적인 인터넷 환경에 잘 부합되는 특징을 가지고 있어서 많은 연구가 진행되고 있다. 멀티 에이전트 시스템에서는 각 에이전트들이 자신의 목적을 위해 행동하기 때문에 에이전트간 충돌이 발생하는 경우에 조정을 통해 협력할 수 있어야 한다. 그러나 기존의 멀티 에이전트 시스템에서의 에이전트 간 협력 방법에 관한 연구 방법들은 동적 환경에서 서로 다른 목적을 갖는 에이전트간의 협동 문제를 올바로 해결할 수 없다는 문제가 있었다.

본 논문에서는 강화학습을 이용한 자동 역할 조정 방법을 통하여 에이전트가 처한 동적 환경에서 서로 다른 목적을 갖는 에이전트간의 협력 문제를 해결한다. 이를 위하여 멀티 에이전트 시스템 분야의 전통적인 문제인 추적 문제에 동적 환경과 서로 다른 목표를 갖는 에이전트들을 모델링하여, 두 가지 수정된 추적 문제를 제안하고 이 문제의 해결을 통하여 제안한 방법이 타당함을 보였다.

연구 세부 분야 에이전트 시스템, 강화 학습, 분산 인공 지능
주제어 에이전트, 강화 학습, 조정

Abstract Agent-based systems technology has generated lots of excitement in these years because of its promise as a new paradigm for conceptualizing, designing, and implementing software systems. Especially, there has been many researches for multiagent system because of the characteristics that it fits to the distributed and open Internet environments. In a multiagent system, agents must cooperate with each other through a coordination procedure, when the conflicts between agents arise, where those are caused by the point that each action acts for a purpose separately without coordination. But, previous researches for coordination methods in multiagent system have a deficiency that they can not solve correctly the cooperation problem between agents which have different goals in dynamic environment.

In this paper, we solve the cooperation problem of multiagent that has multiple goals in a dynamic environment, with an automatic cooperative coordination model using reinforcement learning. We will show the two pursuit problems that we extend a traditional problem in multiagent systems area for modeling the restriction in the multiple goals in a dynamic environment, and we have verified the validity of the proposed model with an experiment.

Keywords Agent, Reinforcement Learning, Coordination

* 본 연구는 '97 한국과학재단 특정연구 (과제번호:97-01-00-08-01-3) 사업지원에 의하여 이루어졌음
** 서강대학교 컴퓨터학과 학·석사 졸업
*** 서강대학교 컴퓨터학과 박사 과정
**** 서강대학교 컴퓨터학과 교수

1. 서론

현실 세계의 문제는 분산되고 개방된(open) 시스템 환경을 기반으로 하고 있다. 개방 시스템은 자체의

구조가 능동적으로 변화할 수 있는 것으로, 서로 다른 사람이 다른 시간에 다른 도구와 기술을 이용하여 만든 매우 이질적인 요소들로 구성되어 있다. 가장 대표적인 분산 개방 시스템인 인터넷은 다양한 종류의 정보가 서로 다른 기관이나 개인들에 의해 생성되어 지리적으로 광범위한 영역에 분산되어 있을 뿐 아니라, 정보 자원이나 통신 링크, 에이전트들의 생성과 소멸은 예측할 수 없다. 이러한 상황에서는 지식과 컴퓨팅 자원, 능력이 제한되어 있는 단일 에이전트를 이용한 문제 해결 방법에 한계가 있다[1]. 이와 같은 환경에서의 문제를 해결하기 위하여 최근 널리 사용되는 방법이 멀티 에이전트 시스템이다. 멀티 에이전트 시스템에서는 에이전트가 대등하게 연결되어 있어 서로 정보를 주고받으며, 또한 서로간에 조정할 수 있는 능력을 가지고 있다.

멀티 에이전트 시스템은 그 연구의 시작을 분산 인공지능(Distributed Artificial Intelligence)에 두고 있으며, 에이전트들 사이의 공동작업을 통해서 각 에이전트의 능력 이상을 요구하는 문제의 해결방법을 찾으려는 연구 분야이다[2][3]. 멀티 에이전트 시스템에 관한 연구는 에이전트들이 해결해야 할 공통의 목표를 설정하고, 각 에이전트들 간의 협력과 조정 과정을 거쳐 주어진 문제를 해결하려는 방식을 사용한다. 멀티 에이전트 시스템에서는 각 에이전트들이 병렬적으로 수행되기 때문에 본질적으로 병렬성을 요구하는 많은 문제 영역에 쉽게 적용될 수 있으며, 이런 병렬성은 특정 부분의 오동작으로 인한 시스템의 전체적인 성능 저하를 방지할 수 있을 뿐만 아니라, 문제가 여러 부분으로 나뉘기 용이할 경우에는 전체 문제 해결 시간을 줄일 수 있다. 또한 전체 문제를 여러 개의 부분 문제로 나누어 각 부분문제를 해결하기 때문에 에이전트의 독립적으로 설계하므로, 중앙 집중적인 시스템과 달리 간단한 프로그래밍 모델을 통한 문제의 해결이 가능하며, 새로운 기능을 가지는 에이전트를 추가함으로써 쉽게 시스템을 확장할 수 있다.

멀티 에이전트 시스템에서 가장 중심적인 연구 과제는 에이전트간의 조정, 협력에 대한 것이다. 에이전트들은 자신에게 할당된 문제를 풀어 나가는 과정에서 부분 관찰(local view), 다중 목표(multiple goal), 분산된 정보 등의 제약 때문에 혼란을 겪거나, 에이전트간의 충돌을 일으킬 수 있다. 또한, 에이전트들이 해결하고자 하는 문제에 대한 제약을 만나거나, 개개의 에이전트가 가지고 있는 각기 다른 능력과 특별한 지식들을 서로 공유할 필요가 있을 경우 에이전트간의 협력이나 조정이 필요하게 되며, 다른 에이전트의 행동에 따라 자신의 행동이 결정될 경우나, 전체 시스템의 효율을 높이기 위해서도 조정이나 협력이 필요하게 된다. 그러나 이를 위하여 제안된 여러 방법들은 각 에이전트들의 역할이 한 번 주어지면 고정되어 동적으로 변하는 개방 환경에의 적용에 적합하지 않다는 문제점을 가지고 있었다.

본 연구에서는 이런 문제점을 극복할 수 있는 에이전트 사이의 역할 조정 모델을 제안한다. 이 모델에서는 에이전트들 사이에 역할 충돌이 일어나는 경우에 강화학습을 이용하여 자신의 역할을 수정한다. 그리고 멀티 에이전트 시스템 분야의 대표적인 문제인 추적문제(Pursuit Problem)[4]에 부분 관찰과 다중 목표의 제약사항들을 모델링하여 두 가지 수정된 추적 문제를 제안하고, 이 문제의 해결을 통하여 제안한 방법의 타당성을 보인다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 기존의 에이전트간 협력 방법에 관한 연구 방법들과 그 제약점들을 살펴보고, 3장에서는 동적인 환경에 적용가능한 역할 조정 모델을 제안한다. 4장에서는 제안한 모델을 수정된 추적 문제에 적용한 실험 결과를 살펴보고 5장에서 결론을 내린다.

2. 기존의 에이전트간 협력방법

기존의 멀티 에이전트 시스템에 대한 연구에서 제한하고 있는 에이전트간의 조정 기술에는, 에이전트의 유기적인 구조를 설계(Organizational Structuring)하는 방법, 에이전트간의 계약(Contracting)을 이용하는 방법, 멀티 에이전트 계획(Multiagent Planning)을 이용하는 방법, 협상(Negotiation)을 이용하는 방법 등이 있다.

우선 에이전트 유기적인 구조를 구성하는 방법은 가장 간단한 조정 방법[5]으로, 에이전트들은 Master / Slave 혹은 Client / Server의 구조를 가지도록 설계한 다음, 에이전트간의 계층적인 관계를 통해 협력과 조정을 수행하게 된다. 이 방법에서 에이전트들은 주로 흑판 구조(Blackboard Architecture)[6]를 이용해서 서로간의 통신을 하게 된다. 그러나 이 경우에는 에이전트들의 구조로 인한 추가적인 제어가 필요하며, 서로간의 통신을 위해서 흑판(Blackboard)을 사용하기 때문에 병목 현상을 발생시키는 등, 멀티 에이전트 시스템의 이점을 저하시킬 수 있으며, 에이전트들이 단순한 구조를 가져야 한다는 제약을

가지게 된다.

계약[7]을 통한 에이전트간의 조정은 주로 Contract Net Protocol(CNP)을 사용하며, 이는 분산된 환경에서 에이전트에 대한 자원 할당이나, 문제의 분배에 주로 사용된다. 우선 관리자로 설정된 에이전트는 자신에게 할당된 문제를 작은 부분 문제들로 나누어 이를 수행할 다른 에이전트를 찾게 된다. 그럼 관리자가 요구하는 문제를 수행할 수 있는 에이전트는 계약자가 되어서 자신이 선택한 부분 문제를 해결하게 된다. 이런 과정이 재귀적으로 이루어지며, 계약자가 되었던 에이전트가 다시 관리자가 되어 문제를 새로이 다른 에이전트에게 할당하게 된다. 에이전트간의 계약에 의한 조정에서는 에이전트간의 계층적인 관계를 자동으로 만들어 내며, 수행할 문제에 대한 자원의 동적 할당이 가능하고, 자연스러운 부하 조절이 가능하다는 장점을 지니고 있다. 그러나, 에이전트간의 조정이 수동적이며, 조정을 위한 통신이 계속 증가하기 때문에 에이전트간에 통신이 이루어지기 어려울 경우 전체적인 시스템 자체에 대한 문제를 불러일으킬 수 있다는 단점을 가지고 있다.

또, 멀티 에이전트 계획을 이용하는 방법에는 중앙 집중적인 계획 방법과 분산적인 계획 방법이 있다[8]. 중앙 집중적인 계획 방법은 하나의 조정 에이전트가 전체 시스템을 관찰하며, 에이전트간의 충돌이 발생했을 경우, 각 에이전트의 계획을 조정하도록 지시를 내려 전체 시스템의 계획을 수정하게 하게 한다. 분산적인 계획에서는 각 에이전트가 다른 에이전트의 계획에 대한 모델을 가지고 있어 서로 통신하며, 작업 수행 중 발생하는 충돌이나 여러 가지 제약사항을 해결하기 위해 각자 가지고 있는 계획을 스스로 수정하게 된다. 이 방법의 경우에는 에이전트가 상당히 많은 양의 정보를 서로 공유해야 하며, 에이전트간의 통신을 위한 추가적인 시간이 필요하게 되어 전체 시스템의 복잡도가 증가하게 되는 문제점을 가지고 있다.

협상을 이용한 에이전트간의 조정은 분산 인공지능에서 조정과 협력을 위한 중요한 방법 중의 하나로 사용되었으며, 기본적으로 에이전트간의 통신을 통해서 상호간에 동의할 수 있는 결과를 이끌어 내는 것을 의미한다. 이 방법에는 게임이론을 기반으로 하는 협상[9], 계획에 기반을 둔 협상[10], 휴리스틱에 의한 협상[11] 등의 방법이 있다. 게임이론을 이용하는 경우 수익행렬을 사용하여, 에이전트의 행동에 따른 이익에 따라 에이전트의 행동을 조정하는 방법을 사용한다. 이 경우에는 에이전트들이 공유해야 하는 서로에 대한 완전한 지식을 가정하기 때문에, 현실세계의 문제에 적용하기 적합하지 않을 뿐만 아니라, 셋 이상의 에이전트의 경우에는 게임이론을 손쉽게 확장할 수 없다는 단점을 가지고 있다. 계획에 기반한 협상의 경우에는 계획에 대한 사전지식에 따라서 에이전트들이 자동적으로 협상을 수행하게 된다. 우선 에이전트는 각자의 행동을 계획하고, 그 계획을 분리된 조정 에이전트가 에이전트의 상태나, 메시지 형태, 대화 방법을 이용해서 협상을 수행하게 된다.

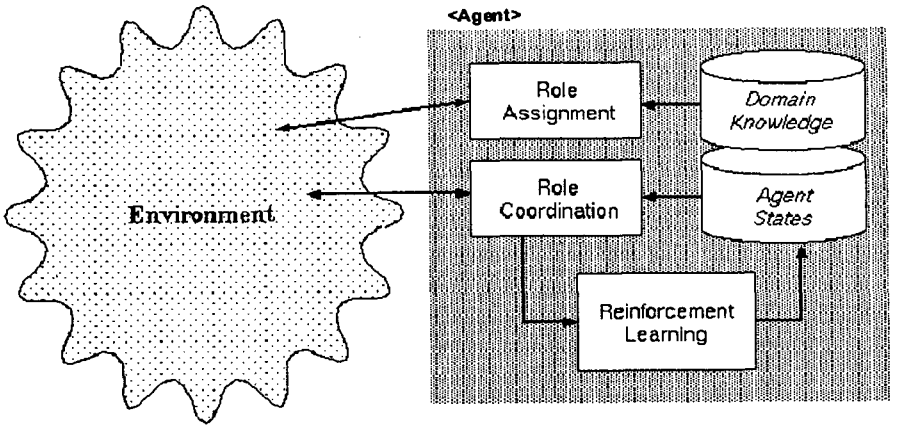
그밖에도, 에이전트간 상호 메시지 전달 없이 사전 협약(Commitment)을 통한 협력 작업을 수행하는 방법[12]이나, 개미나 꿀벌 등 곤충들의 행동 유형을 분석하여 에이전트의 행동 결정에 적용하는 방법[13]들이 있었으나, 모두 각 에이전트들에게 한 번 할당된 목표는 고정되어 변하지 않았기 때문에 에이전트간의 역할에 충돌이 발생하는 경우 이를 해결할 수 없다는 문제를 가지고 있었다.

3. 역할 분담 및 조정 모델

다른 사람과의 협력 과정이 필요한 경우의 예로, 커다란 책상의 위치를 바꾸는 경우를 생각해 보자. 우선 제일 먼저 필요한 일은 책상이 골고루 힘을 받도록 각자의 위치를 결정하는 일이다. 일단 각자의 위치가 결정되고 나면, 이제 원하는 방향으로 나아가기 위하여 앞에서 끌 사람과 옆에서 드는 사람, 뒤에서 밀 사람 등의 역할을 분담할 것이다. 그리고 만약 방향을 바꾸어야 할 경우가 생기면, 서로간의 대화를 통해서 각자 나름대로 역할을 수정하여 원하는 방향으로 전환을 이루게 된다.

본 논문에서 제안한 역할 조정 모델의 구조는 〈그림 1〉과 같다. 이 모델에서는 현재 자신이 처한 환경 상태와 주어진 사전지식을 바탕으로 에이전트의 초기 역할을 결정한 뒤, 자신의 역할을 수행하다가 다른 에이전트와 충돌이 일어날 경우 강화학습에 의한 역할 조정을 하게 된다. 이를 위하여 각 에이전트에게 제공되어야 할 사전지식의 종류는 〈표 1〉에 명시되어 있다.

전체 에이전트 시스템의 목표는 에이전트에 의해 수행되어야 할 역할의 종류를 통하여 설정되며, 에이전트의 가능한 행동의 종류는 각 역할에 대하여 수행하여야 할 행동방식을 통하여 명시된다. 그리고 초기 역할을 결정하기 위한 규칙들과 역할 충돌을 검출하기 위한 규칙들이 도메인 지식으로 주어진다. 따라서, 각 에이전트는 자신이 가지고 있는 이러한 사전지식



〈그림 1〉 역할 조정 모델

〈표 1〉 에이전트를 위한 사전지식

종 류	사 전 지 식
목표 (Goal)	• 각 에이전트에 의해 수행되어야 할 역할의 종류
행동(Action)	• 각 역할에 대하여 수행해야 할 행동 방식
도메인 지식 (Domain Knowledge)	• 자동 역할 분담을 위한 규칙들의 집합 • 역할 충돌을 검출하기 위한 규칙들의 집합

을 바탕으로 결정된 자신의 초기 역할에 따라서 자신의 행동을 수행하여 다른 에이전트와 협력하게 되며, 그 과정에서 다른 에이전트와 충돌이 일어날 경우 강화학습을 이용해 역할 조정을 수행하게 된다.

3.1 Q-learning을 이용한 강화 학습

본 연구에서 제안한 역할 조정 모델에서는 역할 충돌이 발생하는 경우에, 에이전트간 역할 조정을 위하여 Q-learning 방법을 이용한 강화 학습 방법 [14][15]을 사용한다. 강화학습은 시스템의 행동을 관찰하고 그에 대한 적절한 평가만을 제공함으로써, 시스템이 원하는 행동을 나타내도록 학습시키는 방법이다. 특히, 목표를 달성하기 위한 현재 상태의 적합도를 평가하는 가치함수의 추정 방법으로 Q-learning을 사용하게 되면, 학습시킬 시스템에 대한 모델 없이도 온라인 학습이 가능하다. Q-learning 방법에서는 'Q-value'라고 하는 환경 상태와 행동의 쌍을 이용하여, Q-value와 상태 가치 사이의 대응관계를 찾는다. 마찬가지로 가치 함수 대신에 'Q-Function'이라는 용어를 사용한다. 이처럼 환경상태와 행동을 하나의 쌍으로 묶어 사용함으로써, 가능한 모든 행동을 직접 수행해 볼 필요 없이 다음 환경상태에서의 기대값을 구할 수 있게 된다. 뿐만 아니라, Q-learning은 Q-function의 수정을 위하여 단지 한 단계의 수행만을 요구하므로 온라인 학습이 가능하다는 장점이 있다. 이러한 강화학습의 특성들은 동적 환경에서 에이전트간 협력에 필요한 많은 조건들을 만족시켜 준다.

이 논문에서 제안하는 역할 조정 모델에서는 다음과 같이 강화학습 모델을 구성한다.

• 환경상태(Environment State)들의 이산 집합 (S) :

환경상태들의 이산 집합은 역할 충돌이 발생했을 경우에 그 상황을 가장 잘 표현할 수 있는 상태들의 집합으로서, 적용하려는 문제의 특성에 따라 다양한 방법으로 정의할 수 있다.

• 행동(Action)들의 이산 집합 (A) :

역할충돌이 발생하였을 경우에는 조정이 필요한 에이전트가 충돌을 피하기 위해서 취할 수 있는 행동의 종류는 현재 역할의 유지와 다른 역할로의 전이, 두 가지이다. 따라서 사전 지식으로 주어진 역할의 종류가 총 N개일 경우에, 조정이 필요한 에이전트가 나타낼 수 있는 행동의 종류 역시 총 N개가 된다.

• 강화신호(Reinforcement Signal)들의 이산 집합 (i):

강화신호는 다음 두 가지 만을 사용한다.

$\begin{cases} 1: \text{보상} & \text{(역할충돌이 해결된 경우)} \\ 0: \text{처벌} & \text{(역할충돌이 해결되지 않은 경우)} \end{cases}$

조정신호를 받은 에이전트들은 자신이 유지하고 있는 Q-function을 이용하여 새로운 역할을 결정하고, 그 역할에 따른 작업 수행 결과에 따라서 위의 두 가지 신호 중 하나의 강화신호를 받게 된다. 이 논문에서 사용될 Q-Learning의 학습 규칙에서는 Q-function을 이용하여 환경의 상태와 행동을 하나의 쌍으로 묶어 사용함으로써, 가능한 모든 행동들을 직접 수행해 볼 필요 없이 다음 환경상태에서의 기대값을 구할 수 있게 된다.

상태 s 에서 행동 a 를 선택했을 경우의 기대값을 $Q^*(s, a)$라하고, $Q^*(s, a)$의 최대값을 $\max_a Q^*(s, a)$ 라고 한다면, 상태 s 의 상태가치 $V^*(s)$는 아래의 식과 같이 계산된다.

$$V^*(s) = \max_a Q^*(s, a) \quad (1)$$

따라서, $Q^*(s, a)$는 다음 식과 같이 재귀적인 표현으로 나타낼 수 있다.

$$Q^*(s, a) = R(s, a) + \gamma \sum_{s' \in S} T(s, a, s') \max_{a'} Q^*(s', a') \quad (2)$$

여기에서 $R(s, a)$는 상태 s 에서 행동 a 를 수행하여 상태 s' 으로 전이되었을 경우의 보상신호 값을 나타낸다. γ 는 행동 a 를 선택했을 경우, 먼 미래에 나타날 수 있는 영향보다는 현재 환경에 미치는 영향을 더 비중 있게 나타내기 위한 상수값이다. 그리고 $T(s, a, s')$은 상태 전이 함수로써 상태 s 에서 행동 a 를 수행하면 상태 s' 이 된다. 그러므로 Q-Learning 방법의 학습 규칙은 수식(3) 과 같이 표현된다.

$$Q(s, a) := Q(s, a) + \alpha \left(r + \gamma \max_{a'} Q(s, a') - Q(s, a) \right) \quad (3)$$

여기에서 α는 강화 학습의 정도를 결정하는 학습률을 r은 현재 상태에서의 강화신호를 나타내는 값이다.

3.2 강화 학습을 이용한 자동 역할 조정 방법

역할 조정 모델에 기반한 에이전트들은 다음과 같은 단계별 작업 수행 과정을 가진다.

단계 1 각 에이전트는 환경상태(Environment State)와 주어진 사전지식 (Domain Knowledge)을 바탕으로 초기역할을 결정한다.

단계 2 각 에이전트는 부분 관찰 범위 안의 에이전트들에 대하여, 역할 충돌이 발생하는 에이전트가 있으면 조정신호를 전달함으로써 역할 조정이 필요함을 알린다.

단계 3 조정신호를 받은 에이전트들은 가치함수 값에 따라 역할의 변화, 또는 유지 등의 행동을 결정한다.

단계 4 각 에이전트는 현재 역할에 따라 임무를 수행한다.

단계 5 각 에이전트는 이전 단계의 행동 결과를 자체평가하고, 이를 기반으로 강화학습을 수행하여 가치함수를 개선해 나간다.

단계 6 단계 2부터 단계 5까지를 반복 수행한다.

제안된 모델에서는 각 에이전트들은 항상 자신의 부분관찰 범위를 주시하고 있다가, 다른 에이전트가 발견되면 역할 충돌 여부를 판별하여, 역할 조정이 필요하다고 판단되면 해당 에이전트에게 조정신호를 보낸다. 조정신호를 받은 에이전트는 스스로 역할을 변화시키고, 그 결과 역할 충돌의 해결 여부에 따라서 스스로 보상신호를 계산해 낸다. 이러한 과정을 통하여 강화신호에 대한 중앙 집중적인 제어 없이도 강화학습이 가능해 지며 부분 관찰에 대한 제약 역시 포함될 수 있다. 또한 전체 시스템의 목표가 역할의 종류와 각 역할에 따른 임무를 통하여 명시되므로, 각 에이전트는 자신의 역할을 결정하고 그 역할에 따른 임무를 수행함으로써, 전체 시스템이 최종적으로 원하는 목표에 도달할 수 있도록 돕는다. 따라서 각 에이전트는 자신이 처한 환경 하에서 정해진 임무를 수행하기만 할 뿐, 전체 시스템에 대한 평가 정보를 요구하지 않기 때문에 본 논문에서 제안한 강화 학습을 이용한 역할 조정 모델은 에이전트들의 목표가 상충되는 경우 온라인 학습을 통해 역할을 조정함으로써 동적인 환경에 적응할 수 있다는 장점을 가지고 있다.

4. 실험 및 분석

본 연구에서는 역할 분담 및 조정 모델의 타당성을 검증하기 위하여, 멀티 에이전트 시스템 연구에서 대표적인 실험으로 사용되는 추적문제(Pursuit Problem)[3]에 부분 관찰, 다중 목표 등의 제약사항들을 모델링하고, 이를 해결하기 위한 에이전트 시스템을 구성하였다. 추적 문제는 다음의 〈그림 2〉와 같이 격자형의 가상세계와 네 마리의 포식자(Predator), 한 마리의 사냥감(Prey)으로 구성되어 있으며 추적 문제의 목표는 네 마리의 포식자들이 임의의 방향으로 움직이고 있는 사냥감의 주위를 포위하여, 결국 사냥감의 움직임을 봉쇄하는데 있다. 가상세계는 네 가장자리가 연결된 무한의 공간이며, 포식자와 사냥감의 움직임은 가상 공간의 격자를 따라 네 방향으로만 가능하며, 하나의 격자 위에 두 가지 이상의 동물이 동시에 위치할 수 없다. 사냥감의 움직임은 임의대로 정해지며, 포식자에 비해 10%정도 느리다. 포식자들은 서로의 위치를 확인할 수 있다.

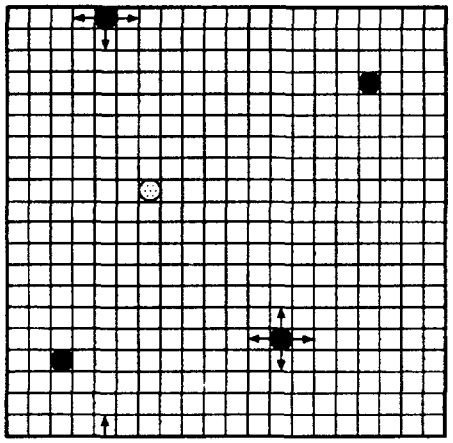
멀티 에이전트 시스템 실험에서는, 각각의 에이전트가 하나의 포식자의 방향을 결정하는 제어기 역할을 수행하며, 이들 에이전트 사이의 협력을 통하여 문제를 해결하게 된다. 본 연구에서는 추적 문제에서 사용되는 멀티 에이전트 시스템에 부분 관찰 문제와 다중 목표 문제를 적용하여, 에이전트간의 역할 조정 모델을 시뮬레이터로 구현하고 그에 대한 실험을 하도록 하였다.

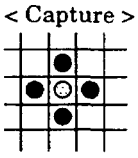
추적 문제 시뮬레이터는 다음의 〈그림 3〉에 나타나듯이 크게 세 부분으로 나뉘어져 있다. 첫 번째 부분은 사냥감과 포식자들이 움직이는 가상 세계를 나타내 주는 영역으로, 30×30 크기의 격자들로 이루어져 있다. 사냥감(Prey)은 항상 흰색으로 표현되며, 포식자(Predator)는 실험에 따라 다르게 나타난다. 두 번째 부분은 가상 세계의 움직임을 제어하기 위한 버튼들이며, 세 번째는 다음의 실험에서 이동 가능 지역이나 모서리까지의 거리를 실시간으로 나타내 주는 그래프 영역이다. 이 시뮬레이터는 Java를 이용해서 구현되었다[16].

실험에서 사용되는 추적 문제는 기본적으로 앞에서 설명된 내용을 따르며 이동형 에이전트가 갖는 제약사항을 모델링하기 위해 부분적인 수정으로 사냥감의 행동방식에 대해서 임의로 움직이던 기존의 방식 대신 좀 더 지능적인 도주를 위하여 이동 가능 영역(Movable Area)을 최대화시키는 방향으로 움직이도록 수정하였다. 이동 가능 영역은 다음과 같이 정의할 수 있다.

"가상 세계의 모든 점들 중에서, 사냥감과의 거리가 다른 포식자들과의 거리보다 짧은 점들의 개수를 이동 가능 영역이라고 한다. 즉, 동시에 출발하여 사냥감이 포식자 보다 먼저 도착할 수 있는 영역을 가리킨다."

이론적으로는 이동 가능 영역을 알아내기 위해서는 가상 세계의 모든 점들을 고려하여야 하지만, 실제로 이동가능 영역을 구할 때에는 계산상의 이득을 위하여 연산 범위를 제한하였다. 이동 가능 영역을 최대

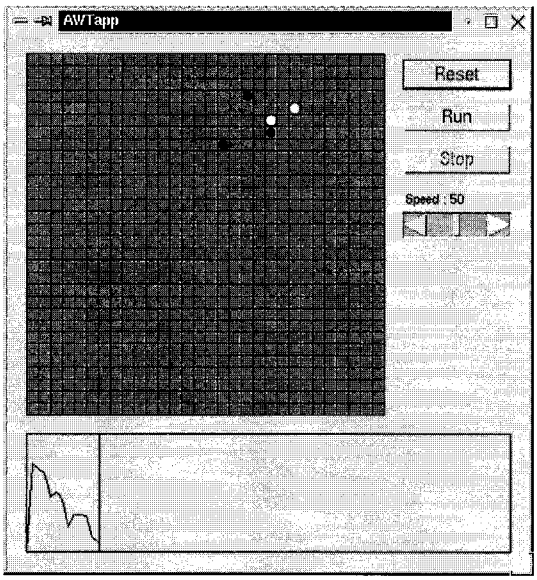




- Predators see each other
- Predators can communicate
- Prey moves randomly
- Prey stays put 10% of time
- Simultaneous movements
- Circulated world

〈그림 4〉 추적문제의 구성 및 목표

화하기 위해서, 사냥감은 우선 이동 가능한 네 가지 방향에 대해 해당 방향으로 이동했을 경우의 이동 가능 영역을 구한다. 그리고 구해진 이동 가능 영역들 중에서 가장 큰 값을 나타내는 방향으로 이동한다. 이처럼 사냥감의 행동방식에 이동 가능 영역의 개념을 도입함으로써, 단순히 현재의 시점에서 포식자로부터 멀어지는 것을 목표로 하는 것이 아니라 앞으로의 여러 가지 가능성을 고려하여 행동을 결정하는 영리함을 보이게 된다.



〈그림 5〉 추적문제 시뮬레이터

4.1 포식자 제어를 위한 에이전트의 설계

기본적으로 추적 문제를 해결하는 아이디어는 매우 간단하다.

"포식자들은 사냥감과의 거리를 줄이기 위하여 움직이되, 포식자들 사이의 거리는 가급적 멀어지도록 유지한다."

기존의 많은 연구들이 위의 간단한 규칙에 기반을 두고 있으나, 멀티 에이전트의 제약 사항을 모델링하는 수정된 추적 문제에서는 포식자의 움직임을 제어하는 에이전트가 다른 에이전트들의 상태나 행동을 관찰할 수 없을 뿐 만 아니라, 모든 포식자들이 무작정 사냥감을 향하여 전진하는 방법으로는 결코 수정된 추적 문제를 해결할 수 없음을 실험결과 알 수 있었다. 따라서, 본 실험에서는 수정된 추적 문제를 해결하기 위해서 포식자를 제어하는 에이전트들을 앞에서 제안한 역할 조정 모델에 기반하여 설계하였다. 다음의 〈표 2〉는 포식자를 제어하는 에이전트가 갖는 사전 지식으로, 프로그램 내에 직접 코드화 되는 정보를 나타낸다.

〈표 2〉 추적문제를 해결하기 위한 에이전트의 사전지식

종 류	사 전 지 식
목 표(Goal)	• 사냥감을 덮치는 방향이 역할이 된다. (EAST, WEST, SOUTH, NORTH의 총 4종류)
행 동 (Action)	• 할당된 역할에 맞는 방향으로 사냥감을 덮친다. (즉, 사냥감의 바로 옆자리를 향하여 이동)
도메인 지식 (Domain Knowledge)	• 사냥감에 대한 자신의 상대적인 위치로 역할을 결정한다. • 포식자가 관찰할 수 있는 최대 길이는 2이다. • 역할 충돌 검출은 다음 규칙에 따른다. 1) 다른 에이전트가 같은 역할을 가진 경우 2) 다른 에이전트가 앞을 가로막아 전진할 수 없는 경우

이 사전 지식을 이용하여 포식자를 제어하는 에이전트간의 역할 충돌이 발생할 경우, 역할 분담 및 조정 모델에서는 강화학습을 통하여 역할 충돌을 해결한다. 포식자 제어를 위한 에이전트는 다음과 같은 강화 학습 모델을 사용한다.

- S = {*EAST*, *WEST*, *SOUTH*, *NORTH*}
- A = {*EAST*, *WEST*, *SOUTH*, *NORTH*}
- i = {0, 1}

여기에서 S, A, i는 각각, 앞의 3.1 절에서 기술한 환경 상태들의 이산집합, 행동들의 이산집합, 강화신호들의 이산집합을 의미한다. 역할 충돌을 해결하기 위한 강화학습 모델을 구성하기 위해서 Q-Function은 4×4크기의 배열을 사용하여 테이블의 형태로 저장하였으며, 3.1 절에서 제안한 수식 (3)의 Q-Learning 학습 규칙을 사용하여 배열의 값을 수정함으로써 역할 충돌을 해결한다.

4.2 〈실험 1〉 부분 관찰 문제

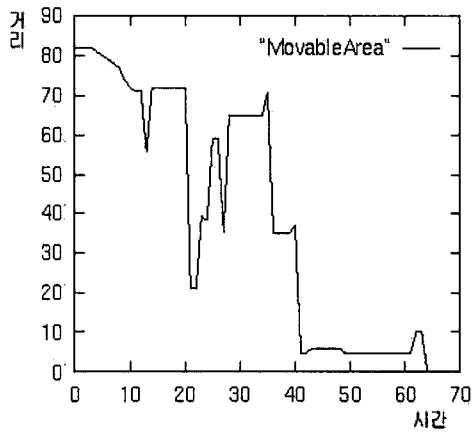
첫 번째 실험에서는 멀티 에이전트가 갖는 제약사항 중에서 부분관찰 문제를 모델링하기 위하여, 본래

의 추적문제에 다음의 〈그림 4〉와 같은 행동제약을 추가하였다.

"각 포식자(Predator)가 관찰할 수 있는 물체까지의 최대길이는 n이다. 즉 각 포식자의 행동을 제어하는 에이전트는 사냥감(Prey)을 제외한 나머지 포식자들에 대해, 거리가 n이하인 경우에만 해당 포식자의 위치나 상태에 관한 정보를 얻을 수 있다."

위의 행동제약을 본래의 추적문제에 포함시킴으로써, 포식자를 제어하는 에이전트가 관찰할 수 있는 시스템의 범위를 전체가 아닌 일부분으로 제한한다. 이는 에이전트가 시스템의 전체를 관찰하는 것을 막을 수 있을 뿐만 아니라, 둘 이상의 에이전트가 서로의 부분 관찰 범위 내에 위치하는 경우에만 강화학습이 수행되도록 함으로써, 멀티 에이전트 시스템에 대한 부분 관찰의 제약을 가할 수 있다.

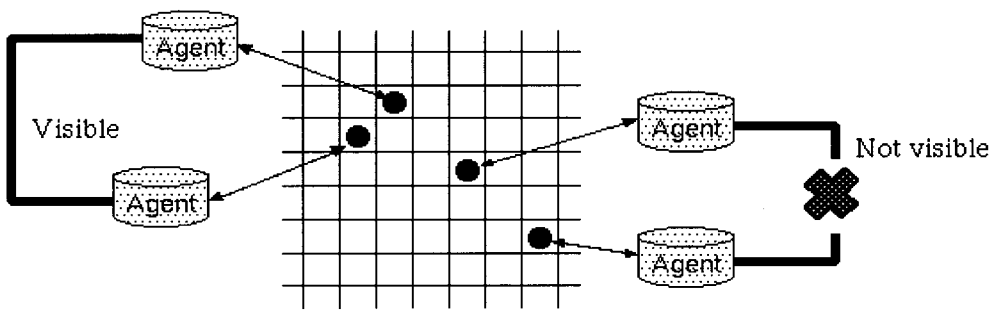
부분 관찰의 제약이 포함된 첫 번째 실험의 결과는 〈그림 5〉와 같다. 〈그림 5〉의 그래프는 사냥감의 시간에 따른 이동가능 영역에 대한 그래프로서, x축은 시간을 y축은 사냥감의 이동 가능 영역을 나타내며, 이동 가능 영역의 변화를 통해서 포식자가 사냥감을 포위하는 과정을 보여주게 된다. 그래프에 따르면, 시간이 지남에 따라서 사냥감의 이동 가능 영역이 점차 줄어들어 결국 사냥감이 포식자들에게 포위되어 이동 가능 영역이 없어지는 것을 보여 주고 있다. 각 포식자들이 초기 역할을 결정하는 시점부터 사냥감을 에워싸는 데까지 걸리는 시간은 각 동물들의 초기 위치에 따라 다양하게 나타나지만, 어떠한 경우라도 결국은 포위 가능하다는 사실을 보여 주었다. 내부적으로 포식자들을 제어하는 각 에이전트들의 행동은 각 포



〈그림 6〉 부분 관찰의 제약이 포함된 추적 문제의 실험 결과

식자들의 초기역할을 결정하고, 이 역할에 해당하는 작업을 수행하며, 역할 충돌이 발생한 경우에는 강화학습을 통하여 서로 간의 역할을 조정하는 과정을 반복하지만, 밖으로 나타난 포식자들의 행동은 초기 상황에서 최대한 신속하게 사냥감을 자신의 반대 방향으로 몰아가면서, 상호작용이 가능한 시점이 되면 포식자간의 세부적인 조정을 통하여 확실하게 사냥감을 포위하는 과정을 보여준다.

〈실험 1〉에서 사용한 추적 문제는 본래의 추적 문제가 본질적으로 포함하고 있는 동적 환경에 대한 제약 사항 뿐 아니라, 실세계의 에이전트가 갖는 부분 관찰(포식자가 관찰할 수 있는 물체까지의 거리에 제약을 둠)에 대한 제약사항을 모델링하고 있으며, 이러한 문제는 본 논문에서 제안한 강화학습을 이용한 역할 조정 모델에 기반한 에이전트간의 협력을 통하여 성공적으로 해결할 수 있음을 보였다.



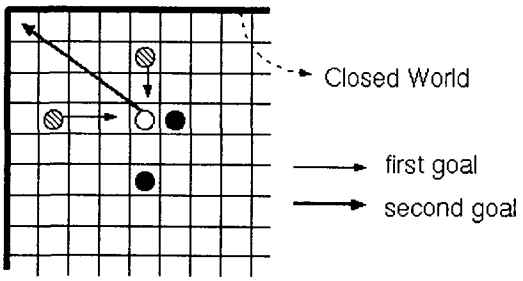
〈그림 4〉 부분 관찰 행동제약

4.3 〈실험 2〉 다중 목표 문제

두 번째 실험에서는 멀티 에이전트가 갖는 다중 목표 문제를 모델링하기 위하여, 〈실험 1〉에서 사용한 추적문제에 다음의 〈그림 6〉과 같은 행동제약을 추가하였다.

"동물들이 존재하는 가상세계는 닫힌 공간이다."

"동일한 가상세계 내에 서로 다른 두 가지 부류의 포식자들이 공존한다. 첫 번째 부류는 다른 포식자들과의 협력을 통하여 사냥감을 에워싸려는 목적을 가지고 있고, 두 번째 부류는 다른 포식자들과의 협력을 통하여 사냥감을 가상세계의 구석으로 몰아넣으려는 목적을 가지고 있다."



〈그림 6〉 다중 목표 행동제약

가상세계에 대한 제약은 한정된 문제의 범위에서 상충되는 두 가지 목표를 모델링하기 위하여 의도적으로 가한 제약 사항으로 '사냥감을 구석으로 몰아넣는다'는 새로운 목표를 가능하게 한다. 위의 두 가지 행동제약으로 인하여, 하나의 가상공간 내에 상충되는 목표를 갖는 두 가지 부류의 포식자가 존재하게 되고, 이들 사이의 역할 조정이 반드시 필요하게 된다. 이는 개방된 환경에서 수행되는 멀티 에이전트가 빈번하게 마주칠 수 있는 상황이다. 〈그림 6〉에서 나타난 것과 같이, 실험에 사용된 가상세계는 쌍으로 이루어진 포식자들로 이루어진다. 첫 번째 쌍은 〈실험 1〉에서 설계한 에이전트 구조를 가진 반면, 두 번째 쌍은 다음의 〈표 3〉과 같은 새로운 에이전트 구조를 갖는다.

새로운 사전 지식을 이용하여 포식자를 제어하는 에이전트의 역할 충돌이 발생할 경우 이를 해결하기 위한 강화학습 모델은 다음과 같이 정의되며, 그 학습 역시 3.1 절에서 기술된 수식(3)에 의해서 수행된다.

〈표 3〉 새로운 형태의 포식자를 위한 에이전트의 사전지식

종류	사전 지식
목표 (Goal)	• 사냥감을 몰아가는 방향이 역할이 된다. (HORIZONTAL, VERTICAL의 총 4종류)
행동 (Action)	• 할당된 역할에 맞는 방향으로 사냥감을 몰아간다. (즉, 사냥감의 바로 옆자리를 향하여 계속 전진)
도메인 지식 (Domain Knowledge)	• 사냥감에 대한 자신의 상대적인 위치로 역할을 결정한다. • 포식자가 관찰할 수 있는 최대 길이는 2이다. • 역할 충돌 검출은 다음 규칙에 따른다. 1) 다른 에이전트가 같은 역할을 가진 경우 2) 다른 에이전트가 앞을 가로막아 전진할 수 없는 경우 3) 사냥감의 앞을 막는 다른 포식자 때문에 전진할 수 없는 경우

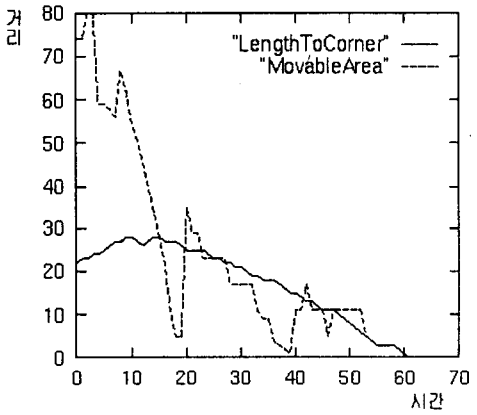
- S = {*HORIZONTAL*, *VERTICAL*}
- A = {*HORIZONTAL*, *VERTICAL*}
- i = {0, 1}

여기에서도 마찬가지로 S, A, i는 각각 앞에서 정의된 강화학습에서의 환경 상태들의 이산집합, 행동들의 이산집합, 강화신호들의 이산집합을 의미한다. 강화학습 모델의 구성은 〈실험 1〉에서 사용한 에이전트와 동일하나, 다른 점은 역할의 종류 개수가 달라졌기 때문에 Q-Function을 표현하는 배열의 크기가 2x2로 바뀌었다는 사실뿐이다.

이 실험의 결과는 〈그림 7〉에 나와 있다. 〈그림 7〉의 그래프에서 실선은 사냥감과 가장 가까운 구석까지의 거리를 나타내며, 점선은 사냥감의 이동가능 영역을 나타낸다. 즉, 그래프의 x축은 시간을 y축은 거리를 의미하며, 그래프를 통하여 각 에이전트 그룹의 행동 방식을 관찰할 수 있다. 실선으로 표현된 사냥감을 구석으로 몰아넣으려는 에이전트 그룹은 사냥감과 가장 가까운 구석 사이의 거리가 지속적으로 감소하는 것으로 보아, 오로지 자신의 역할을 수행하기 위하여 행동하였음을 알 수 있다. 반면에, 점선에서 나타나듯이 사냥감을 포위하는 에이전트들은 자신의 역할을 수행하는 과정에서 다른 에이전트들의 목표를 위하여, 자신의 행동을 조정하였음을 알 수 있다. 〈그

림 7)에 따르면 사냥감을 포위하기 위해 접근하던 에이전트가 자신의 역할을 조정하여 사냥감의 이동가능 영역이 시간 축의 20과 40 근처에서 갑작스럽게 증가함을 관찰할 수 있다. 즉, 사냥감을 포위하려는 역할을 수행하는 것이 아니라, 사냥감을 구석으로 몰아가려는 다른 에이전트 행동을 위해서 자신의 행동을 조정하는 것이라고 할 수 있다.

시뮬레이션의 종료는 사냥감이 가상 세계의 구석에 위치하는 순간에 일어난다. 즉, 사냥감을 에워싸려는 포식자들의 노력은 전체 시스템의 입장에서 보았을 때 장애물에 지나지 않는다. 따라서 구석으로 몰아가려는



〈그림 9〉 다중 목적을 허용하는 추적 문제의 실험 결과

포식자들은 전체 시스템의 목적을 방해하고 있는 포식자에게 조정신호를 보냄으로써, 궁극적으로 사냥감을 구석으로 몰아 가려고 하는 행동에 대한 장애물을 제거할 수 있다. 결국, 구석으로 몰아가려는 포식자는 사냥감을 포위하려는 포식자에 비해서 좀 더 이기적인 성향을 띄게 되고, 사냥감을 포위하려고 하는 포식자는 자신의 행동을 수정하여 다른 포식자가 사냥감을 구석으로 몰아 넣도록 양보해 줌으로써 궁극적으로 사냥감을 구석으로 몰아 넣는데 성공하였다.

본 실험에서는 부분 관찰과 다중 목표 문제를 추가시킨 수정된 추적 문제를 강화 학습을 이용한 역할 조정 모델에 기반한 에이전트 간 협력방법을 통하여 해결 할 수 있음을 보였다.

5. 결론

본 연구에서는 기존의 멀티 에이전트 시스템 분야에서 에이전트간의 역할 충돌이 발생하는 경우 역할의 조정을 통해 이를 해결할 수 있는 역할 조정 모델을 제시하고, 수정된 추적 문제 실험에 이를 적용하였다. 제안된 모델에서는 강화학습을 이용한 역할조정을 통하여 멀티 에이전트 시스템이 처한 동적 환경에서 서로 다른 목표를 갖는 에이전트 사이의 협력문제를 해결하였다. 그리고 수정된 추적문제의 해결을 통하여 본 연구에서 제안한 역할 조정 모델의 타당성을 검증하였다.

추후 수행해야 될 연구로는, 우선 실험으로 사용한 추적 문제가 비록 멀티 에이전트 시스템의 동적인 환경 및 부분 관찰, 다중 목표의 제약사항들을 포함하도록 수정되었다고는 하지만, 실세계의 복잡한 문제들이 갖는 다양한 요소를 모두 포함하고 있다고 할 수 없으므로, 실세계의 다양한 문제를 가지는 응용분야에 대한 적용 실험을 통해 본 연구에서 제안한 방법이 직접 적용될 수 있는지에 대한 연구가 필요할 것이다. 또한, 멀티 에이전트 시스템의 응용분야 중에서 본질적으로 에이전트의 역할의 분할이 용이하지 않는 문제들에 대해서도 자동으로 문제를 세분화하고 에이전트들을 적절한 역할로 나누어 도메인 지식의 의존도를 낮추는 연구도 함께 수반되어야 할 것이다.

마지막으로 강화 학습을 통한 역할 조정방법을 이용하는 경우, 환경상태에 상관없이 무조건 조정신호를 남발하는 불법적인 에이전트에 대한 보안이 존재하지 않으므로 이에 대한 에이전트 시스템의 안정성에 대한 연구도 함께 진행되어야 할 것이다.

참고 문헌

[1] Katia P. Sycara, "Multiagent System", *AI MAGAZINE*, Summer, 1998

[2] Padraig Cunningham and Richard Evans, "Software Agents: A review," *http://www.cs.tcd.ie/Brenda.Nangle/iag.html*, May 27, 1997

[3] H. S. Nwana, L. Lee, and N. R. Jennings, "Co-ordination in Multi-Agent Systems", *Software Agents and Soft Computing, Towards Enhancing Machine Intelligence, concepts and Applications*, Springer, 1997

[4] Peter Stone and Manuela Veloso, "Multi-agent Systems: A Survey from a Machine Learning Perspective," *IEEE Trans. on Knowledge and Data Engineering*, June 1996.

[5] Durfee E. H., Lesser V. R., and Corkill D. D., "Coherent Cooperation among Communicating Problem Solvers", *IEEE Trans. Computers*, 11, pp 1275-1291, 1987

[6] Hayes-Roth B. "A Blackboard Architecture for Control", *Artificail Intelligence*, No 25, pp 251-321, 1985

[7] Smith R. G., "The Contract Net Protocol: High-Level Communication and control in a Distributed Problem Solver", *IEEE Trans. on Computers*, 29, No 12, December 1980

[8] Georgeff M., "A Theory of Action for Multi-Agent Planning". *Proc. 1984 National Conf. Artificial Intelligence*, pp 121-125, August, 1984

[9] David C. Parkes and Lyle H. Ungar, "Learning and Adaption in Multiagent Systems," AAAI workshop on Multiagent Learning, Providence, June 30, 1997.

[10] Kreifelt T. and von Martial F., "A negotiation framework for autonomous agents", in Demazeau Y. and Muller J. P. (Eds): *Decentrilazed AI2, Elsevier Science*, 1991

[11] Werkman K. J., "Knowledge-based model of negotiation using shareable perspectives", *Proc. of the 10th International Workshop on DAI*, Texas, 1990

[12] Sandip Sen and Edmund H. Durfee, "The role of commitment in cooperative negotiation", *International Journal on Intelligent & Cooperative Information Systems*, Vol. 3, No. 1, pp 67-81, 1994

[13] Marco Dorigo, Vittorio Maniezzo, and Alberto Colorni, "The Ant System: Optimization by a colony of cooperating agents", *IEEE Trans. on Systems, Man, and Cybernetics*, part B, Vol. 26, No. 1, pp 1-13, 1996

[14] Leslie Pack Kaelbling, "Reinforcement Learning: A Survey," *Journal of AI Research*, Vol. 4, pp. 237-285, May 1996.

[15] Tom M. Mitchell, *Machine Learning*, McGraw-Hill, 1997

[16] Joseph Kiniry and Daniel Zimmerman, "A Hands-on Look At Java Mobile Agents," *IEEE Internet Computing*, Vol. 1, No 5, pp. 21-30