

멀티 에이전트 시스템에서 협상하는 논리 에이전트에 관한 연구

A Study on Logic Agents that Negotiate in Multi-Agent Systems

이명진(Myung-Jin Lee)¹⁾ 한현관(hyun-kwan, Han)²⁾

요약

멀티 에이전트 시스템(Multi-Agent Systems: MAS)에서 에이전트들이 서로 협상하기 위해서 에이전트들은 공통의 에이전트 통신 언어(Agent Communication Language: ACL)를 사용하여야 하며, 일치에 도달할 수 있도록 설계된 협상 프로토콜을 따라야 한다. 그러므로 에이전트는 위의 요구사항들을 수용할 수 있는 적당한 구조를 가져야 한다. 이 논문에서는 예제에 사용되는 간단한 ACL을 정의하고, 이것과 다른 ACL들(FIPA(Foundation for Intelligent Physical Agents) ACL, KQML(Knowledge Query Manipulation Language))을 간단히 비교한다. 특히, 에이전트를 믿음, 소망, 의도, 그리고 협상 라이브러리로 구성된 논리 프로그램으로 표현한다. 마지막으로, 협상 라이브러리에 포함된 계획자(planner)가 행위들의 계획을 어떻게 제공하는가 그리고 믿음을 어떻게 갱신하는가를 보인다.

Abstract

Agents in Multi-Agent Systems (MAS) should make use of a common Agent Communication Language (ACL) in order to negotiate with others, and conform to negotiation protocols that are designed to reach agreements. Therefore, agents must have suitable architectures that could cover above requirements. In this paper, we define an instructive ACL and compare it with other ACLs such as Foundation for Intelligent Physical Agents (FIPA) ACL and Knowledge Query Manipulation Language (KQML). In particular, we represent agents as logic programs with beliefs, desires, intentions and negotiation library. Finally, we show how the planner, which is in the negotiation library, provides plans of actions and updates agent's beliefs.

Key words: 에이전트 통신 언어(Agent Communication Language), BDI 에이전트(BDI Agent), 논리 프로그래밍(Logic Programming), 협상(Negotiation)

논문접수 : 2004. 12. 10.
심사완료 : 2004. 12. 29.

1) 정회원
2) 정회원

1. 서론

ACL의 주된 목적은 이질적인 에이전트들이 상호 작용할 수 있게 하는 적당한 프레임워크를 모델링하고, 환경과 지식에 관한 정보를 전달하는 의미있는 문장들을 교환하는 것이다[3]. 따라서 ACL은 다음과 같은 바람직한 여러 가지 특징들을 가져야 한다[4]: 에이전트는 자신의 환경과 목표에 대한 지식을 관리하고 사용하기 때문에 ACL은 지식 표현 언어이어야 하며, 선언적이고 구문적으로 간단하여야 하며, 통신 행위에 대한 정확한 의미론(semantics)을 제공하여야 하며, 그리고 네트워크 환경에 적합한 언어이어야 한다. 한편, MAS에서 에이전트들은 그들 자신이 목표를 달성하기에 충분하지 않을 때 에이전트들은 지식을 교환하기 위해 상호 작용한다. 협상 프로토콜은 협상 당사자들 사이의 메시지 흐름을 결정하는데, 이것은 어떤 에이전트가 무엇을 말할 수 있는지, 언제 말할 수 있는지, 그리고 협상 참여자들이 상호 작용하고자 할 때 따라야 할 규칙들을 행위들로 나타낸다[13].

이 논문에서는 다음과 같은 세 가지의 모달리티(modality)들을 고려한다: 환경의 상태에 대한 에이전트의 정신적 태도(mental attitude)를 나타내는 믿음 B , 에이전트의 동기(motivation)를 나타내는 소망 D , 그리고 에이전트의 목표를 나타내는 의도 I . Parsons 예제[11]의 변형을 사용하여 자신의 목표를 달성하는데 도움이 되는 자원을 교환하는 세 개의 협상하는 멀티 에이전트 환경을 가정한다. 에이전트의 지식을 선언적인 방식인 논리 프로그램으로 표현하고, 자원이 한정된 에이전트들이 서로 협상할 수 있도록 이들을 위한 간단한 ACL을 제안하고, 그리고 이것과 몇몇 다른 ACL들(FIPA ACL, KQML)을 비교한다. 마지막으로, 협상과정 동안에 에이전트의 계획자가 행위들의 계획을 어떻게 제공하는가 그리고 믿음을 어떻게 갱신하는가를 보인다.

2. 협상을 위한 에이전트 통신 언어

협상하는 에이전트들 사이의 메시지의 실제적인 교환은 에이전트 자신의 요구, 목표, 혹은 정신적 태도에 의해 유도된다. BDI 에이전트들이 부족한 자원의 할당에 관해 협상한다고 가정하면, 에이전트는 자신의 목표 달성에 필요한 부족한 자원의 할당을 요구한다. 이러한 경우에 자원이 한정된 BDI 에이전트를 위한 통신 언어(communicative act, primitive, 혹은 performative)를 간단히 아래와 같이 표현한다[9]:

```

__ask_if( $a1, a2, m$ ) inform( $a1, a2, m$ )
request( $a1, a2, g, r$ )
__reject( $a1, a2, g, r$ ) give( $a1, a2, r$ )
alternative( $a1, a2, g, [subgoals]$ )
__achieved_goal( $a1, a2$ )
    
```

여기서 $a1$ 와 $a2$ 는 각각 에이전트 식별자, m 은 에이전트의 정신적인 상태에 해당하는 믿음, 소망, 혹은 의도, g 와 g' 은 에이전트의 목표, r 은 자원, 그리고 $subgoals$ 는 목표 달성을 위한 다른 계획을 나타낸다.

위에서 $request(a1, a2, g, r)$ 은 $a1$ 이 자신의 목표 g 를 달성하는데 필요한 부족한 자원 r 을 $a2$ 에게 요청하는 통신 행위인데, $a2$ 는 왜 $a1$ 이 자원 r 을 필요로 하는가를 g 를 통해 알 수 있다. 이러한 정보는 'Kowalski와 Sadri의 observe-think-act 에이전트 사이클'[7]에서 'observe'에 해당하기 때문에, 다른 에이전트에 대한 가설적인(abductive, hypothetical) 추론을 수행할 수 있는 '가설적 논리 프로그래밍 (Abductive Logic Programming: ALP)'[6, 13]에 이용될 수 있을 것이다. 한편, 여기서의 $request$ 통신 행위와 비슷하게 KQML[8]은 $achieve$ (performative)를 제공하고 FIPA ACL[5] 역시 $request$ (communicative act)를 제공하지만, $request$ 통신 행위를 수신한 KQML 에이전트와 FIPA 에이전트는 왜 이러한 요청을 송신 에이전트가 했는가에 대해서는 모른다.

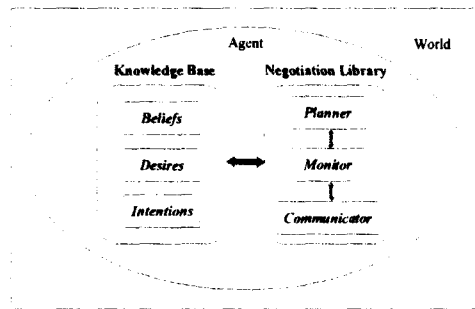
한편, $reject(a1, a2, g, r)$ 은 이전에 $a1$ 이 $a2$ 로부터 수신한 $request(a2, a1, g, r)$ 을 해석한 결과로 생기는 통신 행위들 중의 하나인데, $a1$ 은 자신의 목표 g 을 달성하기 위해 r 이 필요하기 때문에 $a2$ 에게 r 을 넘겨줄 수 없음을 알리는 통신 행위이다. $request$ 통신 행위와 마찬가지로 $reject$ 통신 행위를 통해 수신 에이전트는 상대방 에이전트의 목표 혹은 부목표를 알 수 있다. FIPA ACL $refuse$ 와 $reject-proposal$ 통신 행위 역시 제한한 요청에 거절하는 이유를 'content' 부분에 기술할 수 있다.

3. 협상을 위한 에이전트 구조

BDI 에이전트의 추상적인 특성들을 표현하는 정신적인 상태를 다음과 같이 서술할 수 있다: 세상에 관한 믿음들의 집합, 현재 달성하려는 목표들의 집합, 목표 달성을 어떻게 수행할지 그리고 믿음의 변화에 어떻게 상호 작용할지를 서술하는 계획 라이브러리, 그리고 자신의 목표를 달성하기 위해 수행하려는 계획에 해당하는 현재의 의도[10]. 일반적으로 에이전트는 자신의 목표 달성에 필요한 지식, 작업을 계획하는데 필요한 지식, 그리고 다른 에이전트와 통신하기 위한 지식 등을 필요로 한다. 예를 들어, 에이전트는 문장들의 집합으로 표현될 수 있는 지식을 필요로 한다. 문장들은 자신의 믿음과 능력에 관한 지식, 상호 작용 가능한 다른 에이전트들에 관한 지식, 통신에 필요한 지식, 그리고 특정한 응용 영역에 관한 지식 등을 논리 프로그램으로 표현한다.

이러한 에이전트를 설계하기 위해서 <그림 1>과 같이 지식베이스와 협상 라이브러리로 구성된 협동적인 BDI 에이전트 구조를 고려한다[9]. 여기서 지식베이스는 에이전트 자신의 능력과 다른 에이전트들의 능력에 관한 지식 그리고 문제 해결을 위한 규칙 등에 관한 지식을 포함하는 논리적인 문장들의 집합이다. 지식베이스에 있는 요소들을 에이전트의 정신적인 태도를 표현하는 술어(predicate)들로 표현

한다. 반면에 협상 라이브러리는 계획자(planner), 감시자(monitor), 그리고 통신자(communicator)로 구성되는데, 여기서 계획자는 각각의 작업을 어떻게 해결할지를 결정하며, 감시자는 작업의 실행을 감시하고 메시지를 보낸 에이전트에게 결과를 통보하며, 그리고 통신자는 블랙보드를 사용하는 구조들[1, 2]과는 달리 TCP/IP 소켓을 사용하였다면 에이전트에게 메시지를 보내는 역할 그리고 수신한 메시지를 리다이렉팅(redirecting)하는 역할을 담당한다.



<그림 1> BDI 에이전트 구조

협상하는 에이전트들은 행위에 관한 지식을 공유한다. 에이전트의 계획자는 행위의 선행 조건과 결과 등을 서술하는 집합 B 의 영역에 의존하는 믿음들에 접근하여 목표 달성에 부족한 자원들의 리스트 뿐만 아니라 행위들의 순서 있는 계획을 제공한다. 한편, 자신의 목표를 해결할 수 있는 대안(계획)을 수신한 경우에, 계획자는 가능한 계획들의 영역으로부터 부족한 자원들의 수로 사상하는 비용 함수(cost function)를 사용하여 원래의 계획과 새로운 계획을 비교하여 더 낮은 비용 계획을 선택한다. 여기서 부족한 자원이란 자신의 세계에서 획득할 수 없는 자원을 의미한다 현재 가지고 있는 자원들 그리고 (자신의 믿음 안에 있는) 다른 에이전트가 가지고 있는 자원들은 여기에 해당하지 않음.

한편, 계획자가 정신적인 상태의 참/거짓을

알려주는 *inform*, 자원의 전달을 나타내는 *give*, 그리고 목표 달성을 알려주는 *achieved_goal* 등의 통신 행위를 수신한 경우에, 계획자는 자신의 정신적인 상태를 적절히 갱신하여야 한다(assertion, retraction). 예를 들어, *a1*의 지식베이스에 *b(have(a3, hanger))*가 포함되어 있을 때, *a1*이 *hanger*가 필요하여 *a3*에게 요청하여 *a3*로부터 *give(a3, a1, hanger)*를 수신하였다면 *a1*은 자신의 믿음 뿐만 아니라 *a3*에 대한 믿음을 적절히 갱신하여야 한다.

4. 구현

이 논문에서 제안한 ACL을 자원이 한정된 BDI 에이전트들 사이의 협상 시스템에 적용하여 응용 가능성을 검사한다. 구현은 Java 2 SDK 1.4와 XSB Prolog 2.5를 지원하는 InterProlog 2.0.1로 수행되었는데, 이것은 표준 콘솔 리다이렉션과 TCP/IP 소켓을 사용하여 서브프로세서로 동작하는 Prolog 엔진과 통신하는 Java 응용으로 구성되었다. 한편, 스트림 중심 혹은 메시지 중심의 여러 통신 미커니즘들이 존재하지만 여기서는 소켓을 이용한 버퍼링된 메시지 기반 통신 미커니즘(buffered, message-based communication mechanism)을 사용한다. 즉, 통신 과정은 잘 정의된 한계(boundary)를 가지는 메시지들을 교환한다.

여기서는 다음과 같은 Rao의 공리화를 따른다[12]:

_____goal-intention compatibility: *intend()*
goal()

_____에이전트가 식 를 의도한다면, 달성할 목표로 를 선택하여야 한다.

_____intentions leading to actions:
intend(do()) do()

_____에이전트가 특정한 행위를 수행할 의도를 가진다면, 그 행위를 수행할 것이다.

Rao의공리화에 추가하여, 에이전트가 행위를 수행할 수 있다고 믿을 때 그 행위를 수행하도록 하여 가능한 한 의도를 한정 지우려 하였다

intend(do()) can() do().

_____또한, 믿음에 대한 공리화인 표준 KD45 공리 그리고 MP(Modus Ponens) 추론 규칙에 추가하여아래와 같은 자원이 한정된 BDI 에이전트를 위한공리들을 고려한다:

_____bi(have(X, Z) give(X, Y, Z))
bi(have(Y, Z)). (4-1)

_____bj(have(j, Z) give(X, j, Z)) bj(have(j, Z)). (4-2)

_____bi(have(X, Z) give(X, Y, Z))
bi(have(X, Z)). (4-3)

_____bi(have(i, Z)) bi(holdon(i, Z))
request(X, i, give(i, X, Z)) give(i, X, Z). (4-4)

_____bi(have(i, Z)) bi(holdon(i, Z))
request(X, i, give(i, X, Z)) give(i, X, Z). (4-5)

_____Ii(give(X, i, Z)) *request(i, X, give(X, i, Z))*. (4-6)

_____Parsons 예제의 변형으로 에이전트 *a1*은 그림, *a2*는 거울, 그리고 *a3*는 시계를 걸려고 하지만, 각자의 목표를 달성하기 위해서 *a1*은 nail, *a2*는 hammer, 그리고 *a3*는 hanger_nail이 부족하다. 이제 각각의 계획자는 그 에이전트의 의도를 알아내고, 그 의도를 달성할 목표를 선택하고, 그리고 목표를 해결할 적당한 계획을 결정한다. 예를 들어, *a1*의 믿음들의 집합이 다음과 같을 때,

_____b(have, (a1, hammer)) b(have, (a1, picture))

_____b(have, (a1, screw)) b(have, (a1, screwdriver))

_____b(have, (a1, hanger_nail)) b(have, (a3, hanger))

*a1*의 계획자는 *do(a1, hang_picture)*를 목표로 선택하고 hammer, nail, 그리고 picture를 사용하여 목표를 달성하는 계획을 결정한다. 이제, *a1*의 계획자는 계획의 수행에 필요한 자원들을 검색하여 자신의 세계에서 획득할 수 없는 부족한 자원들의 리스트를 만들어내는데, 이 경

우에 리스트는 [nail]일 것이다. 따라서 a1은 다른 에이전트들(a2, a3)에게 nail에 관한 정보를 물어보게 된다(ask_if(a1, a2, b(have(a2, nail))), ask_if(a1, a3, b(have(a3, nail)))).

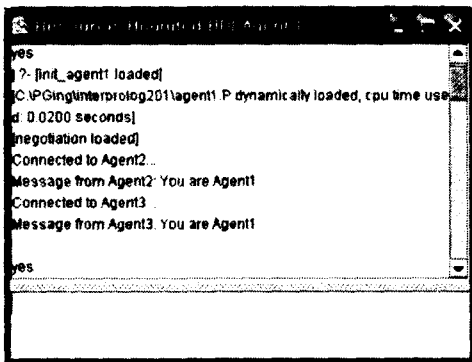
한편, 다음과 같은 믿음들의 집합을 가지는 a2는 hammer, nail, 그리고 mirror를 사용하여 목표를 달성하는 계획을 결정한다.

b(have, (a2, nail)) b(have, (a2, mirror))

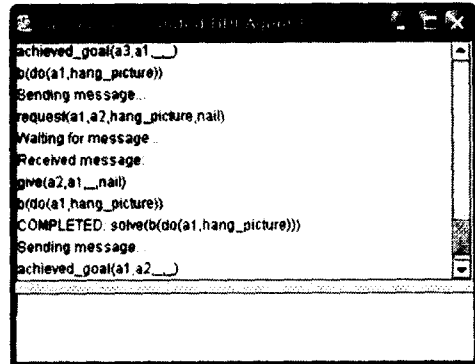
b(have, (a1, screw)) b(have, (a1, hammer))

b(have, (a1, screwdriver))

a2는 목표 달성에 필요한 자원들에 대한 모든 지식을 가지고 있기 때문에, 부족한 자원들의 리스트는 비어있을 것이다. 협상과정 동안에 a2는 alternative(a1, a2, hang_mirror, [screw, screwdriver, mirror]) 메시지를 수신하여 거울을 걸 수 있는 다른 방법을 알게 된다. 이러한 경우에, a2의 계획자는 비용 함수를 사용하여 원래의 계획(p)과 새로운 계획(p)의 비용들(|p|, |p|)을 비교하여 더 낮은 자원 비용 계획을 선택한다. 여기에 관계를 적용시키면 계획자는 새로운 계획을 선택할 것이다. 한편, 에이전트가 식 (4-1), (4-2), 그리고 (4-3)의 발생을 알게(믿게) 될 경우에 자신의 지식베이스를 적절히 갱신한다. <그림 2>는 협상 시작 전의 a1의 상태 그리고 <그림 3>은 협상 종료 후의 a1의 상태를 나타낸다.



<그림 2> 협상 시작 전의 a1의 상태



<그림 3> 협상 종료 후의 a1의 상태

5평가및결론

이 논문에서는 자원이 한정된 BDI 에이전트를 논리프로그램으로 표현하고, 협상을 위한 간단한 에이전트 통신 언어(ACL)를 정의하고, 그리고 협상 계획자가 행위들의 계획을 어떻게 제공하는가에 대해 살펴보았다. 논리 프로그램은 에이전트의 지식을 구문적이면서 선언적으로 표현하고 통신 행위에 대한 정확한 의미를 제공한다. FIPA ACL 그리고 KQML과는 달리, 여기서 제안한 request 통신 행위는 송신 에이전트가 왜 이러한 요청을 했는가에 대해 수신 에이전트가 알 수 있도록 하여 수신 에이전트의 추론 능력을 향상시킬 수 있게 하였다. 특히, 가설적 논리 프로그래밍에서 이러한 정보를 이용하여 가설 추론이 가능하도록 하였다.

행위들의 계획을 만들어내는 계획자는 이용 환경에 따라 STRIPS, situation calculus, 혹은 event calculus 기반 계획자를 사용할 수 있지만, 여기서는 협상 라이브러리에 포함된 계획 라이브러리를 사용하였다. 협동적인 MAS 예제에서 이 계획자는 에이전트의 목표를 결정하고, 목표를 달성할 적당한 계획을 수립하고, 그리고 비용 함수를 사용하여 더 낮은 비용 계획을 선택한다. 비록 이 연구가 제약된 환경에서 수행되었지만, 자원이 한정된 협동적인 MAS에서 이 계획자는 상황에 맞는 적절한 조치를 취하는 것으로 판명되었다.

이 연구에서는 규칙들에 관한 협상을 고려하지 않았지만 에이전트를 설계하고 사용할 규칙들에 대해 협상할 수 있는 동적인 프로토콜에 관한 연구, 보다 개방적인 환경에서 사용될 수 있는 통신 언어와 계획자에 관한 연구, 그리고 전자상거래, 경매 시스템과 같이 일대다수 사이의 협상이 가능한 논리 기반 프레임워크에 관한 연구가 수행되어야 할 것이다.

참고문헌

- [1] C. Bartolini and C. Preist. A Framework for Automated Negotiation. *HP Labs Technical Report 2001-90*, HP Labs Agent Research, 2001.
- [2] C. Bartolini, C. Preist, and N. R. Jennings. A Generic Software Framework for Automated Negotiation. *HP Labs Technical Report 2002-2*, HP Labs Agent Research, 2002.
- [3] B. Chaib-draa and F. Dignum. Trends in Agent Communication Language. *Computational Intelligence*, 18(2), 2002.
- [4] T. Finin, Y. Labrou, and J. Mayfield. Desiderata for Agent Communication Languages, In *Proceedings of the AAAI Symposium on Information Gathering from Heterogeneous Distributed Environments*, 1995.
- [5] Foundation for Intelligent Physical Agents (FIPA). FIPA Communicative Act Library Specification. <http://www.fipa.org/specs/fipa00037/>, 2002.
- [6] A. C. Kakas, R. A. Kowalski, and F. Toni. The Role of Abduction in Logic Programming. *Handbook of Logic in AI and Logic Programming*, 5:235-324, 1998.
- [7] R. A. Kowalski and F. Sadri. From Logic Programming to Multi-Agent Systems, *Annals of Mathematics and Artificial Intelligence*, 25:391-419, 1999.
- [8] Y. Labrou and T. Finin. A Proposal for a New KQML Specification. *Technical Report CS-97-03*, Computer Science and Electrical Engineering Department, University of Maryland Baltimore County, 1997.
- [9] M. J. Lee and J. S. Kim. A Logic Programming Framework for Negotiation among Resource-bounded BDI Agents. In *Proceedings of the International Conference on Intelligent Agents, Web Technologies, and Internet Commerce*, 2001.
- [10] D. Morley. Semantics of BDI Agents and Their Environment. *Technical Note 74*, Australian Artificial Intelligence Institute, 1996.
- [11] S. Parsons, C. Sierra, and N. R. Jennings. Agents that Reason and Negotiate by Arguing. *Journal of Logic and Computation*, 8(3):261-292, 1998.
- [12] A. S. Rao and M. P. Georgeff. Intentions and Rational Commitment. *Technical Note 8*, Australian Artificial Intelligence Institute, 1993.
- [13] F. Sadri, F. Toni, and P. Torroni. Logic Agents, Dialogues and Negotiation: An Abductive Approach. In *Proceedings of the Symposium on Information Agents for E-Commerce*, 2001.