

유비쿼터스 환경에서 멀티 에이전트 시스템을 위한 우선순위 기반 메시지 스케줄링 기법[☆]

Priority-based Message Scheduling for Multi-agent System in Ubiquitous Environment

윤 양 석* 윤 희 용**
Yang Seok Yoon Hee Yong Youn

요 약

유비쿼터스 환경에서 멀티 에이전트 시스템은 지능화 서비스를 제공하는데 유용하다. 본 논문에서는 멀티 에이전트 시스템에서의 통신 성능을 향상시키기 위한 새로운 스케줄링 서비스 모델을 제안한다. 이 모델은 에이전트 간에 교환되는 메시지의 수와 형식화된 ACL 메시지를 기반으로 가중치를 적용하여 우선순위를 할당한다. 제안된 방식은 JADE 플랫폼과 비교하여 메시지 당 처리시간과 총 소요시간을 적게 하는 것을 실험을 통해 보여준다. 더욱이 제안된 방식은 확장성 있는 메시지 처리 시스템을 만들 수 있게 한다.

Abstract

In ubiquitous environment the multi-agent system is effective for providing intelligent services. In this paper we introduce a new scheduling service model for improving the communication performance in the multi-agent system. It is achieved by assigning a priority to the message according to the weight of formalized ACL message and the number of messages exchanged between the agents. An experiment reveals that the proposed approach allows smaller processing time per message and turnaround time than the scheduling with the existing JADE platform. Moreover, the proposed scheme allows a scalable messaging system.

□ keyword : Agent Communication Language, CAL, 부하 분산, 멀티 에이전트, 스케줄링, 유비쿼터스

1. 서 론

유비쿼터스 시스템에서 분산된 에이전트들의 협업을 위한 연구들이 활발히 진행되고 있다[1-3]. 멀티 에이전트 시스템(MAS)에서 에이전트는 동적인 환경에서 다른 에이전트들과 협업을 하는 컴퓨터 프로그램으로 정의될 수 있다. 에이전트는 독립적으로 실행하며 문제를 해결해나가는 프

그램으로 단일 프로그램이 해결하기 힘든 복잡한 서비스들을 제공하기 위하여 다른 에이전트들과 상호 협력을 통해 작업을 수행한다.

에이전트 국제 표준화 그룹인 FIPA(Foundation for Intelligent Physical Agents)[4]에서는 에이전트 간 통신을 위한 에이전트 통신 언어인 ACL(Agent Communication Language)를 제시하였다. ACL의 기본목적은 상호운용성의 보장에 있다. FIPA ACL에서는 인자 중 하나인 communicative act의 집합으로 정의된 프로세스를 제공한다. ACL이나 KQML(Knowledge Query and Manipulation Language)[5]은 다수의 지능적 시스템들 사이에서의 지식과 정보에 대한 교환을 위해 설계되어진 에이전트용 언어이다. KQML에서 정의된 Performative는 FIPA ACL에서의 communicative act와 유사한데,

* 준 회 원 : 성균관대학교 정보통신대학원 컴퓨터공학과
ysyoon@skku.edu

** 종신회원 : 성균관대학교 정보통신공학부 교수
youn@ece.skku.ac.kr(교신저자)

[2007/05/25 투고 - 2007/06/19 심사 - 2007/09/14 심사완료]

☆ 본 연구는 21C 프론티어 연구개발 사업의 일환으로 추진되고 있는 유비쿼터스컴퓨팅 및 네트워크원천기반 기술개발 사업, Post-Bk 사업, 그리고 한국 과학 재단 특정기초기술개발사업 (과제번호: R01-2006-000-10954-0)의 지원에 의한 것임

KQML의 의미를 다른 에이전트에게 알려주는데 사용되어진다. ACL 또는 KQML은 의미적 특징을 가지며 기술적인 정보를 전달하도록 해준다.

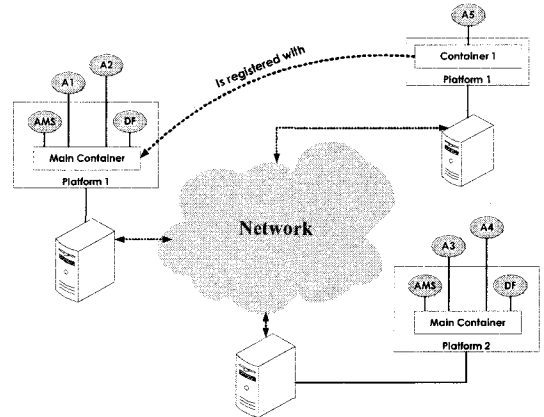
에이전트 프레임워크에서 가장 중요한 요구사항 중 하나는 에이전트들 간의 효율적인 통신을 지원하는 것이다. 실제로 부적절한 스케줄링 정책은 비효율적인 통신을 야기시킬 수밖에 없기 때문에 경량화된 유연한 기능의 스케줄링 모델의 정의가 멀티 에이전트 시스템에서 중요한 이슈가 될 수 있다. 따라서 FIPA ACL에서는 메시지들의 의미를 파악하기 위한 표준을 중요하게 다루고 있다. 본 논문에서는 멀티 에이전트 시스템에서의 새로운 메시지 스케줄링 모델을 제안한다. 제안되는 우선순위 스케줄링 모델의 기본 접근방식은 FIPA communicative act의 의미에 기반을 두어 가중치를 적용하는 것이다. 여기서는 메시지의 우선순위를 결정하기 위하여 형식화된 ACL 메시지들의 의미에 따른 가중치와 에이전트 간에 교환된 메시지의 수에 의거하여 가중치를 적용한다. 메시지의 중요성을 구분하기 위해 채택한 방식은 대화식 메시지들에 높은 우선순위를 할당하는 것이다. 실험을 통해 본 기법이 기존의 JADE 플랫폼에서의 스케줄링보다 메시지 당 처리시간과 소요시간이 작다는 것을 보여준다. 또한, 확장성 있는 메시지 처리 시스템을 가능케 한다.

본 논문의 나머지 부분은 다음과 같이 구성되어 있다. 2장에서는 FIPA 명세서에 기반한 에이전트 통신의 개요를 기술하고 3장에는 제안된 기법을 소개한다. 그리고 4장에서는 실험을 통한 성능평가를 기술하고 마지막 5장에서는 본 논문의 결론과 의견에 대하여 기술한다.

2. 관련 연구

에이전트 플랫폼은 다른 에이전트 플랫폼에 등록된 에이전트들과 협업을 하고 에이전트 관리와 시스템의 검색을 통한 에이전트 그룹의 동적 생성과 삭제에 의한 커뮤니티 컴퓨팅을 지원하기

위한 방법으로 에이전트 통신 언어가 필요하다. 서비스를 요청하는 에이전트에 따라 연관된 그룹들의 동적 생성을 통하여 최적화된 서비스가 제공된다.



(그림 1) JADE 플랫폼

JADE(Java Agent DEvelopment Framework)[6]는 Telecom Italia Lab에서 개발되었고 (그림1)과 같이 에이전트들의 관리 및 모니터링을 제공하고 있는 소프트웨어 프레임워크이다. 여기서는 에이전트간의 통신을 위하여 FIPA 표준을 따르는 에이전트 통신 언어(ACL)를 이용한다. JAVA 언어만을 사용할 수 있는 JADE는 멀티에이전트의 통신을 위한 툴과 패키지를 제공하고 있지만 에이전트간의 대화식 메시지에 대한 고려는 하지 않고 에이전트 통신을 위한 동적 관리가 어렵다.

FIPA[4]는 에이전트 기반 기술을 제공하고 다른 시스템과의 상호 운용성을 지원하기 위한 방안을 제시하고 있다. FIPA ACL 메시지는 하나 이상의 메시지 인자들의 집합으로 구성되어 있는데, 정확하게 어떤 인자가 필요한지 Performative에 따라 효율적인 에이전트 통신 언어를 변경하는 것이 필요하다.

(표 1)에서 볼 수 있듯이 FIPA ACL 메시지 인자들의 집합은 에이전트 간의 통신을 위한 모든 인자들을 포함한다. performative 인자는 ACL 메

시지의 communicative act를 가리킨다. 참여 에이전트를 가리키는 sender, receiver, reply-to 인자는 메시지에 표시되는 참여자를 가리킨다.

(표 1) FIPA ACL 메시지의 인자

Category of Parameters	Parameter
Type of communicative acts	performative
Participant in communication	sender, receiver, reply-to
Content of message	content
Description of content	language, encoding, ontology protocol, conversation-id, reply-with, in-reply-to, reply-by

FIPA CAL은 에이전트가 의도하는 행위를 수행할 수 있도록 의미있는 언어의 형식에 기반하여 작성되어졌다. 이는 Accept Proposal, Agree, Cancel, Call for Proposal, Confirm, Disconfirm, Failure, Inform, Inform If, Inform Ref, Not Understood, Propagate, Propose, Proxy, Query If, Query Ref, Refuse, Reject Proposal, Request, Request When, Request Whenever, 그리고 Subscribe와 같은 라이브러리에 의해 표현된 communicative act로 구성되어있다.

3. 제안하는 메시지 스케줄링 기법

3.1 개요

제안하고자 하는 에이전트 통신 모델은 FIPA CAL의 performative의 의미를 이용하여 개발된다. 여기서는 FIPA communicative act로 형식화된 메시지의 의미에 기반을 두고 대화식 메시지를 구분할 수 있는 방법을 제안한다. 제안된 모델은 미리 정의된 ACL의 CAL을 이용할 수 있게 하는 것에 초점을 맞추었다.

에이전트가 멀티 에이전트 시스템에서 지능적인 서비스의 참여자로서 통신을 하기 위하여

FIPA CAL을 이용하여 정의된 대화식 메시지들은 에이전트의 습성에 대한 패턴으로 존재하게 된다. 우리는 FIPA ACL 인자의 communicative act와 참여 에이전트에 초점을 맞춘다. 멀티 에이전트 시스템에서의 행위의 우선순위는 제공된 형식에 따라 다양하게 정해질 수 있다. 실시간 통신 서비스를 스케줄링하기 위해 제안하는 기본 정책은 다음과 같다.

- 대화식 메시지가 호출되어질 때 질의응답 단계에서의 응답시간을 고려해야한다.
- 모든 메시지는 공평히 취급되어야한다.
- 단위시간당 메시지 전송량은 최대가 되어야한다.
- 메시지의 우선순위는 ACL communicative act로부터 도출해낼 수 있다.

실시간으로 메시지의 통신이 진행될 때 메시지의 중요도에 따라 우선순위가 결정될 수 있는데, 대화식 메시지는 간단한 통보식의 메시지보다 우선순위가 높다. 에이전트들 사이에서의 대화식 행동을 고려할 수 있을 때 그 효율성을 입증할 수 있다. 그러므로 ACL을 사용하는 중요도에 따른 임의의 집합을 제안한다. FIPA에서 제공된 의미론은 에이전트들 사이에서 사회적 관계성을 표시하는 메시지의 효과를 기술하는데 유용하다.

3.2 ACL 메시지의 사회적 약속(Social Commitment)

우리가 제안하는 모델에서는 에이전트들에 의해 교환된 메시지들의 의미에 관련된 communicative act의 형식화된 개념의 의미에 따른 그룹화 전략을 적용한다. 이것은 담화 이론에 따르는데[7-11], (표 2)에서 볼 수 있듯이 지시(directive), 단정(assertive), 허가(permissive), 혹은 금지(prohibitive)로 구분한다. 제안된 기법의 의미론은 송신 에이전트와 수신 에이전트의 관계에 의해 설명할 수 있다.

우리는 대화식 메시지의 가치를 분류하여 세 그룹의 메시지 패턴으로 구분한다. 예를 들면, 하나의 에이전트가 다른 에이전트에게 어떤 주제에 관해 질의하거나 어떤 지능적 행위를 요구하거나 간단한 알림 메시지들을 전달하도록 요청하는 등이 있다. 에이전트는 의미없는 메시지는 전달하지 않기 때문에 모든 메시지들은 표현 또는 응답에 대한 내용을 콘텐츠에 담고 있다.

(표 2) communicative act의 분류된 라이브러리

Patten of CAL	Communicative Acts
지시(Directive)	cancel, call for proposal, propagate, propose, proxy, query if, query ref, request, request when, request whenever, subscribe
단정(Assertive)	confirm, disconfirm, failure, inform, inform if, inform ref, not understood
허가(Permissive) or 금지(prohibitive)	accept proposal, agree, reject proposal, refuse

지시적 담화(The Directive Speech Acts)

제안된 모델의 가장 중요한 관점은 ACL 메시지들을 어떻게 효율적으로 보내느냐는 것이다. 에이전트가 다른 에이전트에 상호작용을 위한 메시지를 보낸다면 빠른 응답이 가능해야한다. 응답 메시지가 도착할 때를 예상할 수 없기 때문에 에이전트는 메시지에 관련된 어떠한 작업도 처리하지 못하기 때문이다. 사회적 담화와 유사하기 때문에 지시적 담화는 단호한 약속(Strong commitment)로 분류할 수 있다. 이것은 요청자 에이전트가 참가자 에이전트에게 표현하는 행위이다. 요청자는 일반적으로 요청하고 명령하며 참가자는 그에 대한 답변을 보내어준다. 대화식 메시지에서 지시적 담화는 어떤 행위를 동반하고 있기 때문에 가장 높은 우선순위를 가진다.

단정적 담화(The Assertive Speech Acts)

단정적 담화는 다른 에이전트에게 어떤 정보를

제공하는데 매우 유용하다. 이것은 또한 통신에 있어서 어떤 목표를 이루고자 하는데 유용하다. 그리고 어떤 행위가 옳거나 그른지에 대한 명백한 정보나 평가에 대한 전달에 관련이 있다. 에이전트의 행위에 있어서 지시적 담화와 유사할 수도 있으며 에이전트가 의도하는 정보를 전달하기 때문에 단정적 담화의 의미는 아마도 단호한 약속 혹은 범용의 약속(Mediocre commitment)로 나타낼 수 있다. 우선순위는 전자 혹은 후자에 따라 양면가치를 지니기 때문에 중간 우선순위를 할당한다.

허가 혹은 금지적 담화(The Permissive and Prohibitive Speech Acts)

허가 혹은 금지적 담화는 찬성 혹은 반대를 결정하기 위하여 사용되어지기 때문에 약한 약속(Weak commitment)으로 분류될 수 있다. 그러므로 가장 낮은 우선순위를 할당받는다.

3.3 대화식 메시지의 우선순위

제안된 모델의 기본 개념은 사회적 약속에 따른 우선순위를 할당하는 것이다. 한편 서비스는 비선점 우선순위 스케줄링(NP-SPS:non-preemptive strict priority scheduling)을 사용한다. 이 스케줄러는 우선적으로 단호한 약속의 대화식 메시지를 처리한다. 그다음 비대화식 메시지가 처리된다. 각각의 메시지는 연속적으로 처리될 수 있도록 허락된 단위 시간량을 할당받게 된다.

하나의 에이전트로부터 다른 에이전트의 메시지 전송의 빈번도는 다음과 같은 참조 행렬에 기입된다.

$$R_{Agent_i, Agent_j} = \begin{bmatrix} null & R_{12} & \dots & R_{1j} \\ R_{21} & null & \dots & R_{2j} \\ \dots & \dots & null & \dots \\ R_{i1} & R_{i2} & \dots & null \end{bmatrix}$$

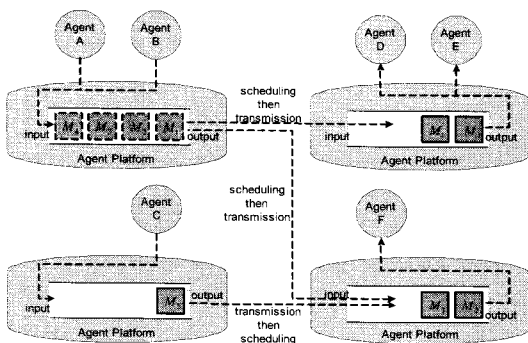
여기에서 R_{ij} 는 에이전트- i 에서 에이전트- j 로 전송되는 메시지의 수를 나타낸다. n 은 시스템에 있는 에이전트의 수이다.

에이전트가 다른 에이전트와 교섭을 시도할 때 스케줄러는 에이전트의 가중치와 우선순위에 의해 계산된 참조 값을 이용한다. 우선순위를 결정하기 위한 처리과정은 다음과 같다.

1. 평가: 메시지의 의미에 따른 가중치를 결정한다.
2. 비교: 메시지의 가중치가 다른 메시지와 같은 가중치를 가지고 있다면 에이전트 간에 얼마나 많은 메시지가 교환되었는지 참조값을 이용하여 우선순위를 결정한다.
3. 정렬: 우선순위를 사용하여 메시지를 정렬한다.
4. 갱신 : 참조값을 갱신한다.

시스템의 대기큐에 기다리는 메시지들보다 높은 우선순위의 메시지가 지속적으로 들어오면 무기한의 연기가 발생되고 시스템의 정지를 초래할 수 있다. 이를 방지하기 위해 제안된 모델에서는 한번 지연된 메시지는 더 이상 지연되지 않도록 한다.

3.4 부하 분산



(그림 2) 복수개의 에이전트 플랫폼에서의 부하 분산

멀티 에이전트 플랫폼에서 비교적 낮은 처리량을 보여주고 있는 에이전트 플랫폼에 많은 자원을 확보해 주기 위한 방법으로 부하분산 모델을

제안한다.

(그림 2)에서와 같이 4개의 에이전트 플랫폼이 존재하고 에이전트 A와 B가 있는 에이전트 플랫폼은 에이전트 D, E 그리고 F가 있는 에이전트 플랫폼보다 높은 처리량을 보여준다고 가정하자. 이 경우 메시지는 보내려는 에이전트의 플랫폼보다 받고자하는 에이전트의 플랫폼에서 스케줄링 처리를 한다. 이러한 방식은 복수개의 에이전트 플랫폼들 사이에서의 스케줄링 서비스에 대한 유연한 부하 분산을 가능하게 해준다. 이것은 에이전트 플랫폼의 주소가 인자에 포함되어있고 메시지의 근원지 및 목적지인 sender, receiver, reply-to 와 같은 인자가 에이전트의 주소를 가리키기 때문에 가능하다.

4. 성능 평가

여기서는 에이전트 플랫폼의 메시지 큐에서의 NP-SPS 스케줄링의 평균 처리시간을 계산하고 JADE와 비교하여 성능 평가를 수행한다. ACL 메시지의 송신자 필드와 수신자 필드는 이름과 주소로 이루어져있고, 콘텐츠 필드는 문자열 4글자로 메시지를 구성하였다. 평가를 위한 송신 에이전트와 수신 에이전트의 실행 수준은 다음과 같다.

- 메시지를 보내기 위한 에이전트의 생성
- 메시지의 생성
- 시작 시간을 기록
- 수신 에이전트에 메시지 송신
- 종료 시간을 기록
- 에이전트의 삭제

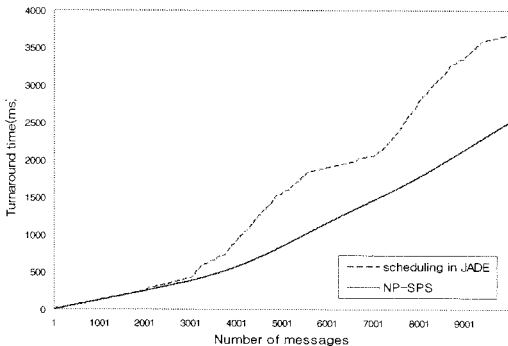
모든 에이전트는 ACL 메시지를 생성하기 위한 약간의 지연 시간을 가지고 순차적으로 생성된다. 우리는 총 측정시간을 전송 메시지의 수로 나누어 평균 처리 시간을 얻는다. 이 평균 처리 시간은 메시지를 보내거나 받을 때 필요한 평균 시간이라고 할 수 있다. 평균 처리시간은 100개의 에이전트들이 두개의 에이전트 플랫폼에서 메시지

를 교환하는데 걸리는 시간을 사용하여 결정한다. 플랫폼의 효율성을 평가하기 위하여 우리는 한 개의 송신 에이전트와 한 개의 수신에이전트로 시작하여 그 쌍을 증가시키면서 처리시간을 측정한다. 에이전트들은 두개의 에이전트 플랫폼에 분산되고 각각의 에이전트들은 10,000개의 메시지를 보낸다. 평균 처리 시간에서 일반적인 에이전트의 관리에 대한 시간은 고려하지 않되 에이전트 생성 및 삭제 시간과 네트워크 부하에 따른 지연 시간을 포함한다.

(표 3) 각각의 메시지 타입에 따른 처리시간

Input message	The average processing time
Strong commitment	0.109462 sec
Mediocre commitment	0.110571 sec
Weak commitment	0.125919 sec
JADE	0.118882 sec

(표 3)에서는 본 연구팀에 의해 개발된 에이전트 플랫폼[2, 3]과 JADE 플랫폼에서의 메시지 평균 처리시간을 보여주고 있다. 제안된 에이전트 플랫폼에서는 대화식 기반의 우선순위 메시지 스케줄링이 적용되었다. 대화식 메시지들인 strong commitment와 mediocre commitment의 메시지들이 JADE 플랫폼에서보다 작은 처리 시간을 보여주고 weak ommitment의 메시지들은 오히려 더 많은 처리시간을 보여준다.



(그림 3) 메시지 처리에 따른 소요시간 (turnaround time)의 비교

(그림 3)은 비교하고자 하는 두 에이전트 플랫폼에서의 소요시간을 보여주고 있다, 여기서 x축은 메시지의 수를 가리키고 y축은 밀리세컨드단위의 소요시간을 가리킨다. NP-SPC 스케줄링의 소요 시간이 JADE 플랫폼의 스케줄링보다 작다는 것을 보여준다. 더욱이 제안된 기법의 총 소요 시간은 메시지가 증가함에 따라 선형적으로 증가하는데 이는 메시지 증가에 따른 확장성에 있어서도 더 우월함을 나타낸다.

5. 결론 및 향후 연구

유비쿼터스 환경에서 멀티 에이전트 시스템을 약결합 분산시스템을 통합하는데 유용하다[12]. 실시간 멀티 에이전트 시스템에서는 에이전트의 효율적인 상호작용을 통한 협업이 필요하다. 본 논문에서는 효율적인 메시지 전송을 위한 스케줄링 기법을 제안하기위하여 FIPA communicative act의 의미에 기반을 둔 가중치 적용 우선순위 스케줄링 모델을 제안하였다. 여기서는 형식화된 ACL 메시지의 가중치에 의해 메시지를 판단하고 에이전트들 사이에서 교환되는 메시지의 수에 비례하여 우선순위를 결정하였다. 대화식 메시지에 높은 우선순위를 부여하였는데, 실험에 의한 성능 평가를 통해 기존의 JADE 플랫폼보다 더 작은 처리시간과 총 소요시간을 허락하는 것을 보여주었고 확장성을 확인할 수 있었다.

향후에는 이러한 연구에 대한 결과를 이용하여 브로커 에이전트와 같은 다양한 에이전트 환경에서의 연구와 실험이 추가로 필요하겠다.

참 고 문 헌

[1] M. Weiser, 'The Computer for the Twenty-First Century', Scientific American, Sept. 1991, pp.94-100
 [2] S. W. Han and H. Y. Youn, 'A Middleware Architecture for Community Computing with

- Intelligent Agents', UbiCNS 2005, June 2005
- [3] S.W. Han, S.K. Song, H.Y. Youn, 'CALM: An Intelligent Agent-based Middleware Architecture for Community Computing', Proceeding of SEUS-WCCIA'06, Oct. 2006, pp.89-94.
- [4] FIPA-Foundation for Intelligent Physical Agents, <http://www.fipa.org>.
- [5] KQML-Knowledge Query and Manipulation Language, <http://www.cs.umbc.edu/kqml/kqmlspec/spec.html>
- [6] JADE, Java Agent Development framework, <http://jade.csel.it>
- [7] J. R. Searle, 'Speech Acts: An Essay in the Philosophy of Language', Cambridge University Press, 1969
- [8] N. Fornara and M. Colombetti, 'Defining Interaction Protocols using a Commitment-based Agent Communication Language', Proceeding of Second International Joint Conference on Autonomous Agents and Multi Agent System', July 2003, pp.520-529
- [9] M. Rovatsos et al, 'Interaction is Meaning: A New Model for Communication in Open Systems', Proceeding of Second International Joint Conference on Autonomous Agents and Multi Agent System, July 2003, pp.536-543
- [10] Guido Boella et al, 'Cooperation and coordination: Role-based semantics for agent communication: embedding of the mental attitudes and social commitments semantics', AAMAS '06, 2006, pp.688-690
- [11] Shakil M. Khan and Yves Lespérance, 'Agent communication: On the semantics of conditional commitment', AAMAS '06, 2006, pp.1337-1344
- [12] Ae Hee Park, So Hyun Park, Hee Yong Youn, 'A Flexible and Scalable Agent Platform for Multi-Agent Systems', CISE 2007, Jan 2007, pp.1-6.

◎ 저자 소개 ◎



윤 양 석(Yang Seok Yoon)

2001년 성공회대학교 정보전산학과(학사)
2005~현재 성균관대학교 컴퓨터공학과(석사과정)
관심분야 : 멀티 에이전트 시스템, RFID, etc.
E-mail : ysyoon@skku.edu



윤 희 용(Hee Yong Youn)

1977년 서울대학교 전기공학과(학사)
1979년 서울대학교 전기공학과(석사)
1988년 Univ. of Massachusetts 컴퓨터공학과(박사)
1988년~1991년 Univ. of North Texas. 조교수
1991년~1999년 Univ. of Texas at Arlington 부교수
2000년~현재 성균관대학교 정보통신공학부 교수 및 유비쿼터스컴퓨팅기술연구소 소장
관심분야 : 모바일 컴퓨팅, 분산처리, 시스템 소프트웨어, etc.
E-mail : youn@ece.skku.ac.kr