

ESB를 사용한 지능로봇의 에이전트 기반 동적 적응'

이재정^o 김진한 이창호 이병정
 서울시립대학교 컴퓨터과학부

jaejeong2@gmail.com {kimjinhan, leechangho, bilee}@uos.ac.kr

Agent based Dynamic Adaptation of Intelligent Robots using ESB

Jaejeong Lee^o Jinhan Kim Changho Lee and Byungjeong Lee
 School of Computer Science, University of Seoul

지능로봇 (Intelligent Robot)은 주변 환경을 감지하는 센서로부터 실시간 정보를 수집하고 지능적인 기능을 수행한다. 이러한 지능로봇은 환경과 동적으로 상호작용하고, 예측하기 어려운 환경에서도 새로운 서비스를 발견하여 스스로 적응해야 한다. 그러기 위해서는 지능로봇에 동적인 적응(dynamic adaptation)이 필요하다. 이러한 소프트웨어 적응 문제를 해결하기 위하여 SOC (Service-Oriented Computing)는 이종의 분산된 서비스 또는 지식을 활용하고 조직하기 위한 주요한 방법을 제공한다. 본 논문에서는 ESB (Enterprise Service Bus)를 사용한 지능로봇의 에이전트 기반 동적 적응 프레임워크를 제안한다. 에이전트 기술은 로봇들이 상호작용하기 위한 지능적인 접근법을 제공하고, ESB는 웹 서비스와 기존의 레거시 시스템 (legacy system) 등을 포함한 다양한 서비스 또는 자원들의 생명주기를 관리하는 효율적인 방법을 제공한다. 본 논문의 프레임워크는 멀티에이전트 시스템을 이용한 서비스 지향 애플리케이션의 동적인 발견과 재구성에 초점을 맞춘다.

본 논문에서는 지능로봇의 동적 적응 프레임워크는 크게 두 부분으로 나뉜다 (그림 1). 먼저 로봇내부의 결정관리자 (decision manager)는 적응을 위한 결정을 내린다. 그리고 재구성관리자 (reconfiguration manager)는 결정관리자의 명령에 따라 재구성을 수행하고 JBI (Java Business Integration) [1]서비스들의 생명주기를 관리한다. JBI 서비스는 레거시 모듈, 웹 서비스 그리고 소프트웨어 컴포넌트와 같은 자원들을 포함한 BPEL (Business Process Execution Language)에 표현된 서비스 조합이다. 프레임워크의 또 다른 부분으로서 로봇 외부의 에이전트들은 서비스 저장소 (service repository)에 서비스를 등록하고 검색하고 순위를 구하기 위해 서로 상호작용한다. 그리고 로봇 시스템을 위한 적합한 서비스를 제공한다. 요청 에이전트 (request agents)는 서비스 설명(service description)을 사용하여 서비스를 요청하고 로봇 시스템으로 결과를 제공하는 인터페이스 에이전트이다. 요청 에이전트는 중개 에이전트로 서비스 검색 요청을 전달한다. 서비스 요청 설명은 적합한 서비스를 발견하는데 필요한 정보를 포함하고 있다. 이러한 정보는 시스템의 요구사항을 표현하기 위한 개념 (concept)과 IOPE (Input, Output, Precondition 그리고 Effect)같은 의미적 정보를 포함한다 [2]. 중개 에이전트 (broker agent)는 요청 에이전트와 제공자 에이전트로부터 요청을 처리한다. 중개 에이전트는 요청 에이전트로부터 서비스 검색 요청을 받고 요청 설명을 이용해서 서비스 저장소에서 적합한 서비스를 검색한다. 그리고 후보 서비스들 중에 가장 적합한 서비스를 선택한다. 서비스 저장소는 서비스와 서비스 속성 정보를 저장한다. 제공자 에이전트 (provider agents)는 서비스 제공자로부터 서비스를 받아 저장소로 서비스를 등록한다. 서비스 제공자는 서비스의 기능적인 정보를 함께 제공해야 한다.

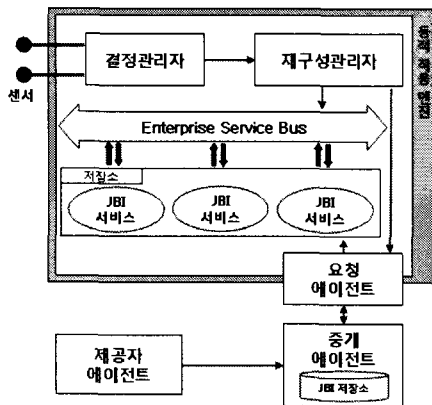


그림 1. 실행시간 발견 및 재구성 프레임워크

에이전트에 기반을 둔 서비스 검색을 위해 먼저 결정관리자는 요청 에이전트에게 서비스 업그레이드를 위한 요청 메시지를 보낸다. 요청 에이전트가 중개 에이전트에게 서비스 검색 메시지를 전달하면, 중개 에이전트는 서비스 저장소로부터 모든 서비스의 목록을 가져온다. 중개 에이전트는 서비스가 적합한지 아닌지를 판단하기 위해 서비스 IOPE와 요청 IOPE를 비교한다. 중개 에이전트는 요청설명에 가중치 값 (weight value)을 이용해서 매칭 정도에 따라 적합한 서비스의 점수 (score)를 계산한다 [3]. 그리고 중개 에이전트는 결정관리자에게 적합한 서비스의 목록을 되돌려준다. 결정관리자는 가장 적합한 서비스 하나를 선택하고 해당하는 서비스를 다운로드를 결정한다. 결정관리자는 요청 에이전트에게 해당 서비스의 컴포넌트 파일을 지정된 폴더에 다운로드할 것을 요청한다. 요청 에이

본 연구는 21세기 프론티어기술개발사업인 인간기능생활지원 지능로봇기술개발사업단의 기술비 지원으로 수행되었음

```

<service>
<name>InfraredApp</name>
<hasInput><direction ?X</hasInput>
<hasOutput><image ?Y</hasOutput>
<hasPrecondition><lux 20</hasPrecondition>
<hasEffect><validImage ?Y</hasEffect>
<textDescription>
(weight{I=1,O=1,P=1,E=1})
</textDescription>
</service>
    
```

그림 2. 서비스 검색을 위한 요청 설명

필요가 있을 때 동적 적용을 위한 계획을 수립하는 것이다. 결정관리자는 이를 위해 몇 가지 정보를 분석한다. 먼저 환경 변화의 정보를 활용하는 규칙 (rule), 상황정보 (context), 목표정보 (goal)를 가진다. 결정관리자는 이러한 정보를 이용해서 현재의 상황을 분석하고 수정한다. 규칙들은 상황정보의 변화로부터 동적인 재구성을 수행할지 안 할지를 결정할 수 있는 정보이다. 결정관리자는 규칙들을 통해 얻은 결정과 이 로봇의 목표정보를 토대로 새로운 계획을 수립한다.

본 연구에서는 ESB를 사용한 지능로봇의 에이전트 기반 동적 적용을 보여주기 위해 로봇 시뮬레이터와 EWS (Extended Web Service)를 구현하였다. 그리고 본 논문에서는 에이전트 구현을 위하여 JADE(Java Agent Development Framework)를 사용하였다. 서비스들은 에이전트들과 UDDI (Universal Description, Discovery, and Integration) 그리고 서비스 저장소로 구성된 EWS에 저장된다. 그림 4는 로봇 시뮬레이터를 보여준다. 로봇 시뮬레이터는 로봇이 업무를 수행하는 동안 예기치 않은 상황에 직면했을 때 EWS의 새로운 서비스를 요청한다. EWS는 요청 설명을 사용해서 UDDI와 서비스 저장소를 검색한다. 그리고 적합한 서비스의 목록을 반환한다. 로봇은 리스트로부터 가장 적합한 서비스를 선택하고 다운로드를 요청한다. 그리고 다운로드된 서비스를 사용하여 재구성한다.

전트는 중개 에이전트를 통해서 서비스 저장소로부터 선택된 서비스 컴포넌트를 얻는다. 중개 에이전트는 XML 문서를 전송하는 표준방법을 이용해서 요청 에이전트에게 서비스 컴포넌트를 전송한다. 그림 2는 서비스 검색을 위한 서비스 요청 설명을 보여준다. hasInput, hasOutput, hasPrecondition, 그리고 hasResult 요소는 서비스의 IOPE를 표현한다. 각 요소의 가중치 값은 textDescription 요소에 표현하며, 요청자의 정확한 환경의 제약을 나타내기 위하여 이러한 가중치 값을 사용한다.

동적 소프트웨어 적용은 시스템을 중지하는 일없이 환경 변화를 감지하고 적응할 수 있는 메커니즘이다. 그림 3은 결정관리자의 내부 모습을 보여준다. 결정관리자의 역할은 주변 환경 변화 정보를 모니터링하고 감지하는 것이다. 그리고 현재 로봇을 재구성 할

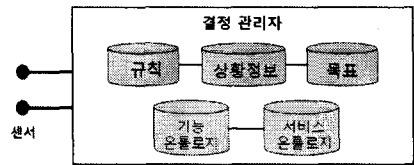


그림 3. 결정관리자 구성도

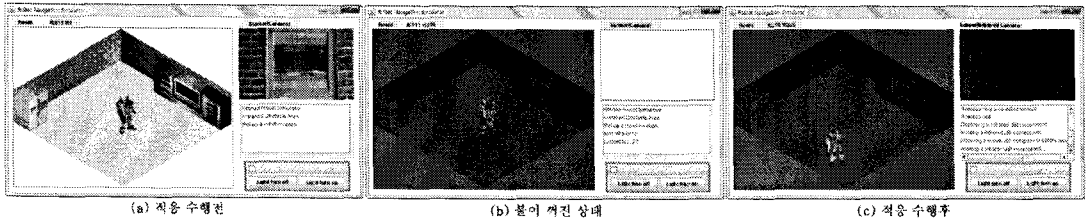


그림 4. 지능로봇 시뮬레이터

본 논문에서는 ESB를 사용한 지능로봇의 에이전트 기반 동적 적용 프레임워크를 제안하였다. 하지만 본 프레임워크는 지능로봇과 UDDI/서비스 저장소간의 네트워크가 인터넷을 통해서 연결되기 때문에 실행 시 성능의 제한을 가진다. 향후의 연구에서 로봇이 특수한 환경에 적응할 때 지능로봇의 성능을 개선하기 위한 연구가 수행되어야 할 것이다. 또한 상황정보를 가지고 추론(reasoning)을 통해서 로봇의 행동을 예측함으로써 로봇의 성능을 개선할 필요가 있다. 이러한 목적을 달성하기 위해 앞으로 체계적으로 로봇의 상황정보를 분석하고 로봇의 행동을 결정하는 연구가 필요하다.

참고 문헌

[1] J. ji-chen and G. Ming, "Enterprise service Bus and an Open Source Implementation," Proc. Of International Conference on Management Science and engineering, pp. 926-930, 2006.
 [2] S. Chaiyakul, K. Limapichat, A. Dixit and E. Nantajeewarawat, "A Framework for Semantic Web Service Discovery and Planning," Proc. of IEEE Conference on Cybernetics and Intelligent Systems, pp. 1-5, June 2006.
 [3] J. Kim, J. Lee, B. Lee, " Runtime Service Discovery and Reconfiguration using OWL-S based Semantic Web Service," Proc. of IEEE 7th International Conference on Computer and Information Technology, Oct. 2007.