

시맨틱스 기반 소프트웨어 관리 기법 분석¹⁾

이화연*, 최호진, 고인영
한국정보통신대학교 공학부
{leehy*, hjchoi, iko}@icu.ac.kr

Semantically based Software Description Model

Hwayoun Lee*, Ho-Jin Choi, In-Young Ko
School of Engineering, Information and Communications University

요 약

다양한 환경에서 지능적으로 적응하는 소프트웨어를 위한 연구는 활발히 이루어지고 있으나 구체적인 응용 분야에 적용되지 못하고 있는 형편이다. 본 논문에서는 로봇과 같이 적응형 소프트웨어 기술이 필요한 분야에서 시맨틱스 기반의 소프트웨어 관리 기법을 적용하는 방법을 분석한다.

1. 서론

동적인 환경에서 적응적으로 대응할 수 있는 소프트웨어 기술에 대한 연구는 국방 분야, 우주탐사 분야와 같이 실시간에 매우 안정적으로 기능을 수행해야 하는 응용분야들을 중심으로 주로 진행되어 왔다. 그러나 예측할 수 없는 상황에 대한 대처나 적응 기술은 현재의 소프트웨어 기술로는 해결이 어려워 지속적인 연구개발이 필요한 부분이라 할 수 있다. 특히 지능로봇과 같이 다양한 기능을 제공하면서 인간의 일상생활과 밀접한, 일반화된 시스템에서의 적응형 소프트웨어 기술이나 자가 치유 및 개선 기술들에 대한 연구는 아직 큰 진전이 없는 상태로 국내 뿐 아니라 국외에서도 많은 연구가 시작되는 단계라 할 수 있다[1]. 특히 시맨틱스(semantics) 정보를 소프트웨어 구성에 활용하여 보다 지능적으로 필요한 컴포넌트들을 찾고, 의미 있고 유연한 소프트웨어를 만드는 기술은 구체적인 응용분야에 직접 적용되어 실험되지 못했다.

본 논문에서는 기존에 수행되었던 시맨틱스 기반 모델 연구를 분석하여 지능로봇과 같이 적응형 소프트웨어 기술이 필요한 분야에서 활용될 수 있는 컴포넌트 관리 기법의 방향을 제시한다.

2. 배경

시맨틱스 기반의 소프트웨어 조지기법은 DARPA(Defense Advanced Research Projects Agency)의 지원 하에 DASADA(Dynamic Assembly for System Adaptability, Dependability, and Assurance)[2] 프로그램 내의 프로젝트들에 의해 그 기반 기술의 연구가 이루어졌으나 실제의 문제에 적용, 실험되지 못하여 응용분야에 보다 적합한 시맨틱스 표현 및 추론 방법들이 제시되지 못하였다. DASADA 프로그램은 2002년경부터 그 다음 단계의 연구 프로그램 (DASADA Phase II)이 시작되어 실제 응용분야로 적용하는 과정이 수행될 계획이었으나 사정에 의해 이루어지지 못한 것이 주요 요인이다.

또한 온톨로지 기반의 소프트웨어 시맨틱스 추론 방법은 카네기멜론 대학과 몇몇 기관에서만 제한적으로 연구되었고, 이러한 연구들도 응용분야가 정해지지 않은 상태에서 이론적인 연구로만 그쳤다. 특히 카네기멜론 대학에서 개발한 인터넷 에이전트의 능력 비교를 위한 시맨틱스 추론 방법은 입출력 자료의 형태나 값의 범위 등을 제한하는 것과 같은 아주 구체적이고 특별한 상황에 기반하고 있으며, 시맨틱스가 일치 하지 않을 경우 대체 컴포넌트를 찾거나

1) 본 연구는 대학 IT연구센터 육성지원사업의 연구결과로 수행되었음

변환기를 삽입하는 등의 보다 진보적인 연구 및 적용은 이루어지지 못하였다.

3. 로봇 기능의 수집/증식을 위한 학습 기술

본 장에서는 로봇 기능의 수집 및 증식을 위한 기술의 구현을 위해서 시맨틱스 기반의 소프트웨어 상호 운영성 및 호환성을 측정하는 기법과 부가 및 대체 컴포넌트 추론 기술, 컴포넌트의 수집 및 증식 기술에 대하여 분석한다.

3.1 시맨틱스 기반의 소프트웨어 상호 운영성 및 호환성 측정 기술

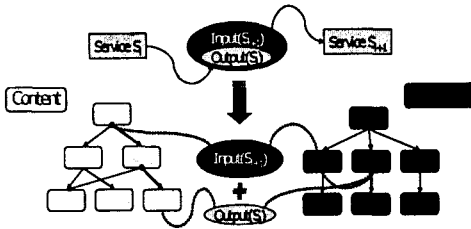


그림 1. 온톨로지 기반 컴포넌트간의 상호 운영성 판별법

로봇이 네트워크 환경을 통하여 다양하고 많은 소프트웨어 컴포넌트에 접근할 수 있을 때, 목표로 하는 소프트웨어 시스템에 맞는 컴포넌트를 비교하여 찾아내는 것은 중요한 문제이다. 단순히 컴포넌트의 이름이나 입출력 데이터의 표현 구조 (Syntax or Data Structure) 만을 고려하는 기존의 방법을 사용하면 신뢰적이고 안정적인 시스템을 구성하는 것이 어렵다. 따라서 컴포넌트들의 기능 및 입출력 정보의 시맨틱스를 파악하여 목표로 하는 소프트웨어의 기능을 구현하기 위한 올바른 기능을 제공하고, 정확하게 상호운영 될 수 있는 컴포넌트만을 소프트웨어의 조직과정에 포함시키는 방법이 필요하게 된다.

본 논문에서는 온톨로지를 기반으로 컴포넌트들 사이의 시맨틱스 관계를 파악하여 상호간의 운영성 및 호환성을 판별하는 기법을 분석한다. 즉 그림1에서 보듯이 여러 가지의 온톨로지 기술된 입출력 데이터의 시맨틱스를 온톨로지의 포함관계 (Subsumption Relation)를 통해서 두 컴포넌트의 입출력 간 호환이 가능한지를 파악하는 것이다. 어떠한 시맨틱스를 갖는 입력 데이터를 받는 컴포넌트 (S_{i+})는 그것의 입력보다 구체적인 시맨틱스를 갖는 데이터를 출력하는 컴포넌트 (S_i)와 입출력 관계를 통해 상호 연결될 수 있다.

이러한 간단한 시맨틱스 정보의 비교 원리에서 출발하여 로봇이 보다 정확하고 지능적으로 자신의 소프트웨어를 재구성하고 성장해 나가기 위해서는 다음과 같은 세부 기술들이 연구되어야 한다.

(1) 온톨로지 기반 시맨틱스 거리 측정 기술

이 기술은 온톨로지 구조 내에서 두 지점 간의 거리를 계산하여 소프트웨어 속성 사이의 관계를 정량적으로 알아내어 소프트웨어 컴포넌트 간의 상호 운영성 및 호환성을 정확하게 판단하기 위한 것이다. 일반적으로 두 소프트웨어 컴포넌트 간의 상호 운영성 및 호환성을 판단하기 위해서는 한 가지 온톨로지만 고려해서는 부족하며, 여러 소프트웨어 속성들을 표현하는 온톨로지 정보를 복합적으로 고려하여 상호 관계를 파악해 내는 기술이 필요하게 된다 [3,4]. 그리고 이러한 다중 온톨로지를 복합적으로 고려해서 컴포넌트 간의 시맨틱스 거리를 측정할 수 있는 소프트웨어 측정기 (Software Gauges)의 개발도 필요하다.

또한 네트워크 환경에서 접근 가능한 공적인 소프트웨어 컴포넌트를 이용하기 위해서 시맨틱 웹 (Semantic Web) 등에서 제안하는 표준적인 시맨틱스 표현법 및 추론법을 적용하는 것이 필요하다.

로봇이 보다 신뢰적으로 자신의 소프트웨어를 재구성하고 성장시켜 나아가기 위해서는 소프트웨어 컴포넌트들의 기능, 입출력 뿐만 아니라 그들의 신뢰성, 성능, 자원이용도 등을 고려한 시맨틱스 정보를 추론하여 이용하는 것이 필요하다. 이러한 컴포넌트 외적인 특성들도 고려하여 컴포넌트 간 상호 운영성 및 호환성을 판단하는 기법의 개발이 요구된다.

(2) 복수 입출력 변수들의 시맨틱스를 고려하여 소프트웨어 컴포넌트 간의 상호 운영성 및 호환성을 판별하는 방법의 성능 증대

컴포넌트 사이에 비교해야 할 입출력 변수의 개수가 증가함에 따라 시맨틱스 추론에 걸리는 시간이 기하급수적으로 증가하는 문제가 발생한다. 따라서 컴포넌트 간에 고려해야 할 시맨틱스 조건들이 많을 때 이러한 성능저하 문제를 해결하기 위하여 컴포넌트 간의 효율적인 시맨틱 바인딩 (Semantic Binding) 알고리즘이 필요하다. 고려할 조건들의 중요도에 따라 우선순위를 두어 시맨틱스를 비교하는 등의 방법을 실험하여 가장 효율적인 시맨틱 바인딩 방법론을 찾아야한다.

특히 학습기능과의 연계를 통해 서로 바인딩 했던

사례들을 기록하고 이용하여 바인딩 추론성을 개선해야 한다. 그리고 소프트웨어 아키텍처 내에서의 물리적 상호 연결을 고려하여 컴포넌트 간 바인딩 관계를 추론함으로써 시맨틱스 기반 컴포넌트 간 바인딩의 성능과 정확도를 높이고자 한다.

3.2 부가 및 대체 컴포넌트 추론 기술

로봇이 필요로 하는 컴포넌트를 찾아 자신의 소프트웨어에 삽입할 때 기존의 컴포넌트들과의 상호 운영성의 문제가 발견되면 그것을 해결하기 위해 부가적인 컴포넌트들을 식별하여, 취득, 삽입하거나 기존의 컴포넌트를 다른 것으로 대체하여 주는 기술이 필요하다. 시맨틱스를 기반으로 이러한 부가적인 또는 대체할 컴포넌트들을 효율적으로 찾는 방법을 개발하기 위해서는 다음과 같은 세부 연구가 요구된다.

(1) 두 컴포넌트간의 간접적 연결에 필요한 중간 컴포넌트 추론 기술

두 컴포넌트가 상호 연결되어야 할 때 둘 사이에 시맨틱스의 차이가 있으면 이를 해결하기 위하여 부가 컴포넌트들이 식별되어 삽입되어야 한다. 또한 시맨틱스는 호환이 되나 문법적 (Syntactic)인 차이가 있어 실질적으로 컴포넌트간의 데이터 교환이 불가능 할 때는 해결을 위해 부가적인 데이터 변환기 등을 삽입해야한다.

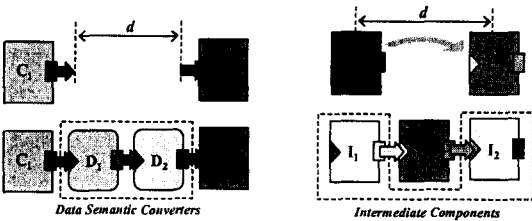


그림 2. 시맨틱스 기반 컴포넌트 간 상호 운영성 및 호환성 측정

그림2에서 왼쪽 그림은 이러한 시맨틱스의 차이, 그에 따른 부가 컴포넌트의 삽입 과정을 도식적으로 보여주며, 오른쪽 그림은 어떠한 컴포넌트 (C₁)를 다른 컴포넌트 (C₂)로 대체해야 할 경우 서로간의 시맨틱스의 차이를 측정하는 절차를 보여준다.

(2) 목표로 하는 소프트웨어 재구성에 필요한 컴포넌트들을 추론하는 기술

로봇이 문제해결을 위해 소프트웨어 재구성의 필요

성을 판단하였을 때, 현재의 소프트웨어와 목표로 하는 소프트웨어 간의 시맨틱스 차이를 분석하고, 이러한 차이를 극복하기 위한 부가 컴포넌트 조합을 생성하는 기술이 필요하다. 이 기술은 앞에서 설명한 부가 컴포넌트 추론기능을 반복적으로 적용하여 구현한다. 또한 전체 컴포넌트 조합의 일관성 유지를 위한 여러 가지 기법의 연구가 뒤따라야 한다.

그림3은 소프트웨어가 재구성 될 때 부가적으로 필요한 컴포넌트를 온톨로지 기반의 추론법을 사용하여 찾아내는 과정을 보여준다.

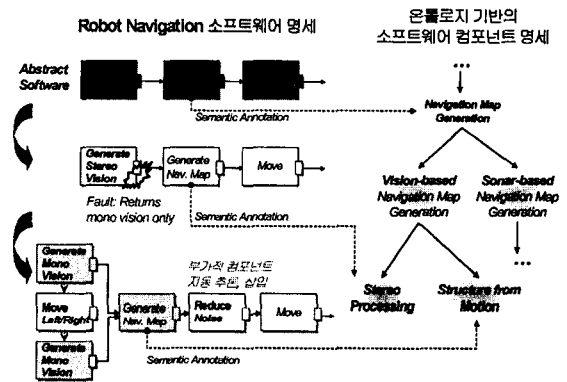


그림 3. 소프트웨어 재구성시 부가 컴포넌트의 추론 과정

(3) 소프트웨어 아키텍처 재구성시 필요한 부가 컴포넌트 추론 기술의 개발

로봇의 문제 해결이나 자가 성장을 위해 소프트웨어 아키텍처를 재구성해야 하는 경우 기존의 아키텍처에서 변경되는 부분들을 추적하여 컴포넌트들 간의 새로운 연결이 의미적으로 올바른지를 판단해야 한다. 이렇게 아키텍처의 재구성을 통한 소프트웨어 변화를 추적하고 소프트웨어 전체의 올바른 시맨틱스를 유지하기 위한 기법들이 필요하다. 또한 소프트웨어 아키텍처로부터 추상적 소프트웨어 구성을 도출하는 기법을 개발하여 아키텍처와는 독립적으로 소프트웨어 시맨틱스의 정확성을 추론해야 한다.

3.3 컴포넌트 수집 및 증식 기술

로봇이 문제해결에 필요한 컴포넌트 및 부가적인 컴포넌트를 찾아 이용함에 따라, 이러한 경험들을 수집하여 저장하여 로봇 자신의 문제 대처 능력을 키우는 방법이 필요하다. 이렇게 컴포넌트 취득 및 사용 상황들을 수집하여 저장하고, 차후에 비슷한

문제가 발생하였을 때 집의를 주면 이러한 정보를 찾아내어 그와 유사한 방식으로 새로운 문제를 해결해야한다.

(1) 시맨틱스 기반의 컴포넌트 정보 저장소

새로운 컴포넌트에 관한 정보를 수집 저장하기 위해서는 시맨틱스 기반의 컴포넌트 정보 저장소가 필요하다. 이러한 저장소 모델을 설계하여 새로운 컴포넌트 정보가 입력 될 때 온톨로지 구조 내에 새로운 컴포넌트를 삽입할 적절한 위치를 판별할 수 있어야한다.

또한 저장된 컴포넌트들의 시맨틱스 정보를 효과적으로 검색하기 위한 질의 언어를 개발하고, 질의를 처리하여 컴포넌트 정보를 검색하기 위한 방법을 개발해야 한다. 그리고 컴포넌트의 사용 빈도, 문제 발생 빈도 등도 수집, 기록하여 컴포넌트 선택 시 고려함으로써 로봇이 보다 더 신뢰적인 컴포넌트를 찾아 사용할 수 있도록 한다.

(2) 공용 컴포넌트 저장소 (Public Component Repository)와의 연계기술

로봇의 외부 네트워크 환경에 존재하는 다양한 소프트웨어 컴포넌트들을 문제해결을 위해 이용하기 위해서는 표준 기반의 컴포넌트 저장소 접근 기술이 구현되어야 한다.

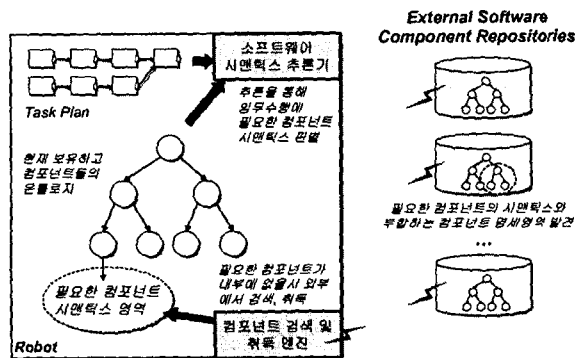


그림 4. 외부 공용 컴포넌트 저장소 접근을 통한 컴포넌트 증식

그림4는 소프트웨어상의 문제해결을 위해 필요한 컴포넌트가 로봇 내부의 로컬 컴포넌트 저장소에 없을 때 로봇이 외부의 공용 컴포넌트 저장소를 접근하여 필요로 하는 컴포넌트의 시맨틱스 부분과 일치되는 컴포넌트 및 그 온톨로지 정보를 취득하여 증

식하는 과정을 보여준다.

4. 결론

본 논문은 시맨틱스 기법을 적응형 소프트웨어 컴포넌트의 기능 수집 및 증식을 위하여 적용하는 기법에 대하여 분석하였다. 구체적인 기법으로 시맨틱스 기반 소프트웨어 상호 운영성 및 호환성 측정 기술, 시맨틱스 기반 대체, 부가 컴포넌트 판별기술, 컴포넌트의 수집 및 증식 기술에 대하여 기술하였다.

본 논문의 목표인 적응형 소프트웨어에서 컴포넌트를 관리하는 기법 분석은 현재와 같이 다양해지고 복잡해지는 컴퓨팅 환경에서 반드시 필요한 기술이다. 이 기술은 다양한 시스템에 응용가능하며, 이를 통해 관련 산업의 동반적 기술 상승효과를 기대할 수 있다.

5. 참고문헌

[1] In-Young Ko, Robert Neches, Ke-Thia Yao "A Semantic Model and Composition Mechanism for Active Document Collection Templates in Web-based Information Management Systems", ETAI, Vol. 5, Section D, pp. 55-77, 2001

[2] DARPA DASADA Web Site: <http://schafercorp-ballston.com/dasada/>

[3] Sandra Heiler, Renée J. Miller, and Vincent Ventrone "Using Metadata to Address Problems of Semantic Interoperability in Large Object Systems" First IEEE Metadata Conference, Silver Spring, Maryland, April, 1996.

[4] Sergey Melnik et al. "Generic Interoperability Framework" Working Paper, Department of Computer Science, Stanford University. <http://www.diglib.stanford.edu/diglib/ginf/WD/ginf-overview/>