

URC를 위한 명령 분석 및 서비스 계획 기술

Command Analysis & Service Planning Technology for URC

목 차

- I . 서론
- II . 자연언어 기반 인간-로봇 인터페이스
- III . 상황인지를 위한 컨텍스트 처리기술 동향
- IV . 서비스 컴포지션 기술 동향

김재홍 (J.H. Kim)	지식및추론연구팀 선임연구원
하영국 (Y.G. Ha)	지식및추론연구팀 선임연구원
박천수 (C.S. Park)	지식및추론연구팀 연구원
장민수 (M.S. Jang)	지식및추론연구팀 선임연구원
이미경 (M.K. Lee)	지식및추론연구팀 연구원
윤여훈 (Y.H. Yoon)	지식및추론연구팀 연구원
손주찬 (J.C. Sohn)	지식및추론연구팀 선임연구원, 팀장

로봇 지능은 다양하게 해석될 수 있으며 해석에 따라 다른 요소기술들을 포함하게 된다. 본 고에서는 '사용자의 명령과 주어진 상황을 이해하여 수행할 서비스 목표를 설정하고 이를 달성하기 위한 일련의 작업들을 계획'하는 데 필요한 지적 능력을 로봇 지능으로 정의한다. 이와 같은 로봇의 지능적 서비스 수행 능력을 실현하기 위한 기본적인 요소 기술에는 자연언어 기반의 인간-로봇 인터페이스, 컨텍스트 처리, 서비스 컴포지션 기술이 포함된다. 본 고에서는 이들 요소 기술의 현황과 로봇에의 응용 방향을 살펴본다.

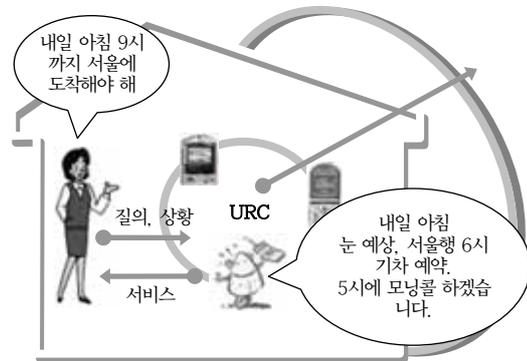
I. 서론

초기 로봇은 컴퓨터와 기계의 결합을 통하여 재프로그래밍이 가능한 범용 조작기의 개념으로 정의되었다. 로봇 관련 기술이 발전되어 가는 과정에 있어 지능의 중요성이 강조되었고 감지 및 인지기술, 인간-로봇 상호작용 기술, 자율주행 및 조작 기술 등이 요소기술로 부각되었다[1]. 로봇이 인간과 공존하면서 지능적인 서비스를 제공하기 위해서는 인간의 명령을 로봇이 정확히 이해하고 합리적인 상황 판단을 통해 명령을 충족시키기 위해 일련의 작업들을 계획하고 실행하는 과정이 필요하며, 본 고에서는 이러한 과정에 필요한 요소기술들에 대해 살펴본다.

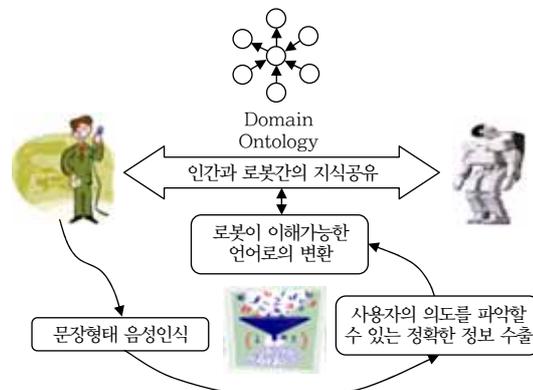
II장에서 다루는 자연언어 기반 인간-로봇 인터페이스는 전통적인 인간-로봇 상호작용 기술의 한 분야로 생각할 수 있고, III장에서 다루는 상황인지는 감지 및 인지기술과 연관성을 가진다. 또한, IV장에서 다루는 서비스 컴포지션 기술은 주행, 조작 등을 그 대상으로 할 수 있다. 이와 같이 본 고에서 다루고 있는 기술들은 전통적으로 지능로봇에서의 요소기술들과 밀접한 관련을 가지며, 사용자의 명령을 지능적으로 처리하기 위해 필수적인 것들이라고 할 수 있다.

II. 자연언어 기반 인간-로봇 인터페이스

자연스러운 인간-로봇 상호작용은 사람과 로봇의 육체적, 정신적 또는 감성적인 상호작용을 포함한다. 그 중 정신적 상호작용의 경우 로봇이 인간의 사고를 도와주고 자극할 수 있는 대화를 통한 상호작용 기술이 요구된다. 현 단계의 인간-로봇 상호작용 연구는 시각 정보처리 기술, 음성 정보처리 기술 그리고 감성 및 인지관련 기술 등의 요소기술에 집중되고 있다. 음성정보처리는 오랜 역사를 가지고 있는 기술로서 전화를 이용한 정보검색 서비스, 휴대폰 서비스, 번역기 등 다양한 분야에 상용화되고



(그림 1) 말귀를 알아 듣는 로봇 개념도



(그림 2) 말귀를 알아 듣는 URC 인터페이스를 위한 요소 기술

있다. 인간-로봇 상호작용 분야에서의 음성정보는 사용자의 위치 파악 및 명시적인 명령어 입력 수단으로 주로 사용되고 있다[1].

(그림 1)과 같이 사용자의 숨은 의도까지 포함한 사용자의 질의에 대한 서비스를 제공하는 말귀를 알아듣는 로봇의 구현이 자연언어 기반 인간-로봇 인터페이스의 궁극적인 목표라고 할 수 있으며, 이를 위해 (그림 2)와 같은 요소기술이 필요하다. 본 고에서는 이들 요소기술과 이들 요소기술들을 활용한 인터페이스 기술에 대해 살펴보기로 한다.

1. 문장형태 음성인식

음성인터페이스는 인간과 기계간의 상호작용을 위해 가장 자연스러운 인터페이스 중 하나로 로봇

분야에서도 필수적인 요소이다. 현재 개발된 대부분의 로봇에는 음성인터페이스가 제공되며, URC를 위해 ETRI 음성인터페이스 연구팀에서 2006년까지 단문명령, 2007년 제한된 대화명령, 2009년 자유대화명령에 대한 인식을 가능하도록 하기 위한 계획을 가지고 있다[2]. 본 연구팀에서는 말귀를 알아듣는 자연언어 기반 URC 인터페이스 개발을 위해 상기의 음성인식엔진을 사용하고 있으며 2005년 초 현재 실험실 수준에서 사용 가능한 낭독체의 단문인식이 가능한 상태이다.

2. 인간과 로봇간의 지식공유

지식공학 분야의 태스크, 문제해결방법(PSM) 및 추론 등과 로봇 분야의 로봇 아키텍처, 외부 환경 모델링, 플래닝 및 제어 등은 상당한 공통점을 가지고 있다고 할 수 있으며, 공통의 온톨로지를 통해 지식공학에서의 기법들을 로봇 시스템에서도 적용 가능하다. 현재 로봇 분야에서는 개발의 중요한 부분들이 특정 구현에 상당히 의존적이기 때문에 결과물 중 많은 부분이 태스크나 로봇 플랫폼 또는 환경이 바뀌면 다시 사용되기가 어렵다. 따라서, 반복적인 실험이나, 아키텍처의 재사용 등이 어렵고 궁극적으로 지식 모델 및 구현에 대한 검증 및 평가를 위한 적절한 도구의 사용이 어려워지게 된다[3]. 로봇 시스템을 지식기반시스템과 로봇 하드웨어 플랫폼의 결합이라는 관점에서 보면 지식공학에서 사용하는 방법론을 로봇 분야에서도 사용 가능하며, 이것은 이들간을 연결해 줄 수 있는 공통의 온톨로지에 의해 가능해진다고 할 수 있다.

3. 정보추출

구글이나 야후 등에서 제공되는 서비스인 정보검색이 대용량의 문서집합으로부터 사용자의 질의에 맞는 문서집합을 찾아내는 것으로 정의한다면, 정보추출은 사용자의 관심사에 적합한 문서들로부터 어떤 구체적인 사실이나 관계를 정확히 추출해내는 작업을 의미한다. 즉, 정보 추출은 주어진 문맥 내에

서 사용자가 원하는 정보를 선별적으로 골라내는 작업을 의미한다.

정보 검색을 중심으로 연구되어 온 정보 처리 기술은 사용자가 필요로 하는 정보만을 추출하는 맞춤형 정보 검색 기술과 다양한 지식원 및 대용량 정보 자료로부터 특정 주제에 관한 정보를 추출하여 체계적으로 관리하는 방향으로 발전되고 있다. 사용자 맞춤형 정보 서비스, 지식 관리 시스템의 구축 등 유용한 정보 자료만을 선별하기 위한 목적으로 미국에서는 1990년대 초반부터 MUC와 TREC을 중심으로 정보 추출 및 정보 검색 회의를 개최해 왔다. 또한, 1999년에는 일본어에 대한 정보 추출 회의로 IREX가 개최된 바와 같이 국외에서는 오래 전부터 정보추출에 관한 연구가 활발히 진행되고 있다[4].

자연언어 기반 URC 인터페이스에서의 정보추출은 음성인식의 결과로 넘어온 명령문에서 로봇이 서비스를 제공하기 위해 필요한 정보들을 추출해 내는 과정이다.

4. 자연언어 기반 URC 인터페이스 시스템

본 연구팀에서는 음성인식기능과 정보추출기술을 지식공학적 접근방법에 접목시키는 인터페이스에 대한 연구를 진행하고 있다. (그림 2)에 나타난 것처럼 이와 같은 접근방식이 가능하게 하기 위해서는 기존의 방식보다 복잡한 구조를 가지게 된다. 우선, 입력 음성은 음성인식을 통해 문자열로 변환되며, 전통적인 방식에서 다소 변형된 정보추출 기술에 의해 로봇이 서비스를 수행하기에 충분한 만큼 중요한 정보들을 추출한다. 추출된 정보는 단어 형태로 된 것이므로 이것을 로봇에서 이해 가능한 언어로 변환한다. 이때, 로봇언어의 기본 단위는 온톨로지의 개체(entity)가 되며, 단어를 개체로 변환하기 위해 온톨로지 서버와의 통신이 필요하다.

이러한 구조가 가능하기 위해 다음과 같은 사항들이 요구된다. 첫째, 음성인식기는 문장단위의 인식이 가능해야 한다. 둘째, 정보추출기는 로봇과의 대화를 위해 충분한 의미 정보를 문장에서 추출해야

한다. 셋째, 사람과 로봇과의 연결 (또는 시스템 내의 다른 컴포넌트들과의 통신을 위한, 더 나아가서 로봇간의 통신을 위한)을 위한 온톨로지를 구축하고 있어야 하며, 온톨로지 서버가 이에 대한 질의에 응답이 가능해야 한다. 넷째, 로봇 언어가 정의되어야 하며, 다섯째, 자연언어를 로봇언어로 변환할 수 있는 방법이 있어야 한다. 본 연구팀에서는 제한된 도메인에서 이와 같은 과정을 모두 포함하는 시스템 프로토타입 구축을 완료한 상태이다.

Ⅲ. 상황인지를 위한 컨텍스트 처리기술 동향

1. 컨텍스트의 정의

컨텍스트는 사람, 장소, 객체 등 사용자와 애플리케이션 간의 상황을 특징지을 수 있는 정보를 말하며 사용자와 유비쿼터스 컴퓨팅 환경 사이에 관련된 사용자의 환경, 오브젝트, 상태에 관한 모든 정보를 말한다[5]. 컨텍스트라는 용어는 여러 분야에서 다양한 의미로 사용되고 있으나, Xerox PARC의 B. Schilt는 컨텍스트를 “사용자와 오브젝트에 관련된 신원 및 오브젝트 정보” 정의하였고, GATECH의 G.D. Abowd는 “사용자, 공간, 오브젝트 등의 개체와 관련된 모든 정보”라고 정의하였다[6].

컨텍스트의 종류로는 위치, 시간과 같은 물리적 컨텍스트, 날씨, 밝기, 소음 등의 환경적 컨텍스트, 주식 시가나 스포츠 점수 등의 정보적 컨텍스트, 건강, 기분, 스케줄, 활동성 등의 개인적 컨텍스트, 그룹활동이나 사회적 관계 등을 나타내는 사회적 컨텍스트, 웹사이트를 방문하거나 전자메일을 받는 것 같은 애플리케이션 컨텍스트, 네트워크의 정보나 프린터 상태 등을 알 수 있는 시스템 컨텍스트 등으로 나눌 수 있다[7].

2. 로봇과 컨텍스트 인지

로봇에서 사용자 중심의 지능적인 서비스를 제공

하기 위해서는 사용자 및 사용자 주변환경에 대한 정보를 분석하여 사용자의 요구를 파악하는 컨텍스트 인지 기술이 중요한 역할을 담당한다. 지능형 로봇 분야에서 컨텍스트 인지 기술을 이용하면, 사용자의 명령에 대해 사용자의 상태, 물리적인 환경, 기존 정보를 통한 분석 등의 상황정보를 이용하여 사용자의 상황에 맞는 결과 값을 제공하게 된다. 사용자 ID, 위치, 시간, 온도, 심리적인 요소, 사용자의 선호도 등의 상황 정보를 이용하게 되면 로봇이 사용자의 상황과 변화에 적응하는 지능적인 서비스를 제공할 수 있다.

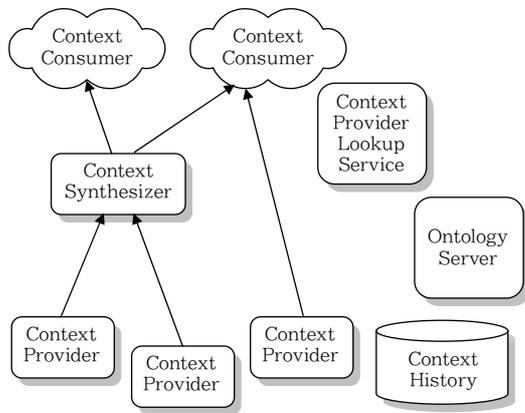
이제 컨텍스트 인지 기능을 제공하는 대표적인 두 가지 시스템을 통해 컨텍스트 처리 기술 현황을 살펴보고자 한다.

3. 컨텍스트 인지 프레임워크 기술 동향

가. Gaia

Gaia[8] 프로젝트는 Illinois 대학에서 개발한 분산 미들웨어로, 유비쿼터스 컴퓨팅 환경의 스마트 스페이스에서 이용되는 컨텍스트 인지 에이전트(context-aware agent)를 제공한다. 컨텍스트 처리를 위해 논리 추론과 기계 학습 방법이 폭넓게 활용되며, 서로 다른 유비쿼터스 컴퓨팅 환경뿐만 아니라 서로 다른 에이전트간의 시맨틱한 상호 운용성을 보장하기 위해서 DAML로 기술된 온톨로지를 사용한다. Gaia상의 모든 에이전트는 CORBA 환경 위에서 작동되며, 이를 위해 CORBA의 다양한 미들웨어 서비스가 활용된다[9].

컨텍스트 인지 프레임워크의 개념적인 구조는 (그림 3)과 같다. 컨텍스트 인지 프레임워크는 탐색, 컨텍스트 감지(sensing), 이벤트 분산 등의 서비스를 수행하는 여러 에이전트들로 이루어져 있다. 이 모델에서는 각종 센서들과 컨텍스트 정보를 제공하는 데이터 소스를 컨텍스트 제공자(context provider)라고 부르며, 여기서 수집된 데이터들은 컨텍스트 합성기(context synthesizer)에 의해 고수준의 추상적인 컨텍스트로 추론된다. 컨텍스트



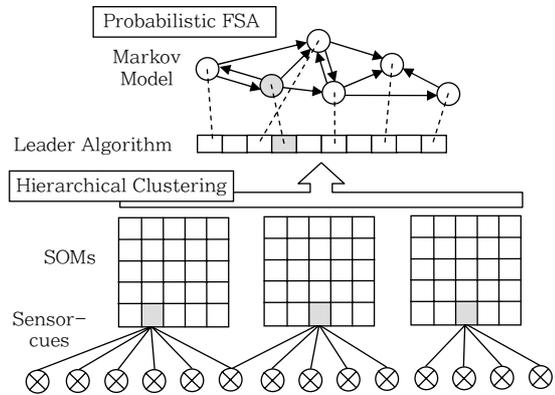
(그림 3) Gaia의 컨텍스트 인지 모델 구조

소비자(context consumer)는 컨텍스트 합성기나 컨텍스트 제공자에서 제공하는 컨텍스트를 기반으로 상황을 인지하고 인지된 상황에 적합한 서비스를 제공하게 된다. 컨텍스트 제공자 룩업 서비스(context provider lookup service)는 컨텍스트 소비자가 필요로 하는 컨텍스트를 가져올 때, 적합한 컨텍스트를 제공하는 컨텍스트 제공자를 찾는 역할을 하며, 컨텍스트 히스토리(context history)는 과거의 컨텍스트를 저장하는 역할을 한다. Gaia의 컨텍스트 인지 프레임워크에는 에이전트들 간의 시맨틱한 상호 운용성을 위해 사용되는 온톨로지를 관리할 수 있는 온톨로지 서버(ontology server)가 포함되어 있다.

나. TEA

TEA는 유럽연합의 지원을 받아 노키아(Nokia) 등 네 개 기관이 함께 약 2년 여간 진행한 과제이다. TEA는 소형 이동 장비에 장착 가능한 컨텍스트 인지 프레임워크 개발을 목표로 하였다. 소형 이동 장비는 TEA를 통해 상황을 파악하고 이로부터 사용자가 원하는 서비스를 자동으로 찾아낼 수 있다[10].

컨텍스트 인지 프레임워크의 개념적인 구조는 (그림 4)에 보는 바와 같다. 각종 센서들로부터 수집된 센서값들은 그 종류에 따라 간단한 전처리 과



(그림 4) TEA의 컨텍스트 인지 모델 구조

정을 거쳐 중간 단계의 Cue로 변환된다. 이렇게 생성된 Cue들은 Cue의 개수를 길이로 하는 특징 벡터(feature vector)를 형성하게 되며 이 벡터는 다차원 공간 상의 한 점을 가리킨다. TEA에서 컨텍스트 인지는 이 Cue 벡터들을 특정한 컨텍스트로 분류(classification)하는 작업을 통해 이루어진다. TEA에서 컨텍스트는 현재 상황을 기술하는 기호값(symbolic value)으로 정의된다. TEA에서는 Cue 벡터의 분류를 위해 SOM을 사용하고 있다.

TEA의 컨텍스트 인지 프레임워크는 상기한 바와 같이 패턴 분류(pattern clustering)에 기반한 인지 방법 외에 일련의 의미있는 컨텍스트 변화를 감지할 수 있는 기술을 포함한다. (그림 4)의 최상위에 표시된 확률 오토마타(probabilistic finite state automata) 부분이 바로 컨텍스트의 상태 변화를 학습하는 기능을 담당한 부분이다. 이 기술을 이용하면 과거의 컨텍스트 변화와 현재의 컨텍스트를 기반으로 미래의 컨텍스트를 예측할 수도 있다.

TEA의 컨텍스트 인지 프레임워크는 기계 학습 기법을 응용함으로써 인지 능력의 적응성(flexibility)을 확보하는 동시에 최적의 실행 성능을 제공하는 아키텍처로 설계되었다. 상황 인지에 있어서 일련의 의미있는 상태 변화로부터 한차원 높은 상황 지식을 추출할 수 있는 기능이 제공되는 점도 우수한 점이라 사료된다.

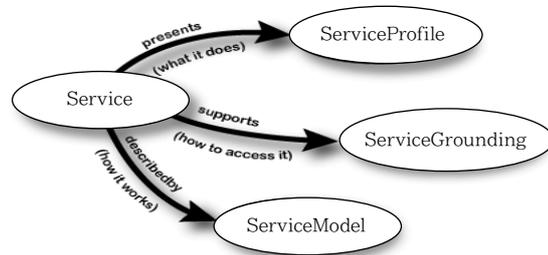
IV. 서비스 컴포지션 기술 동향

현재 웹 서비스[11] 기반의 서비스 컴포지션 기술에 대한 연구는 여러 연구그룹에서 진행되고 있다. 대표적인 연구로써 Maryland 대학의 SHOP2 [12] 기반의 웹 서비스 컴포지션 시스템이 있는데, 이는 HTN[13] 기반의 AI 플래닝 기술을 적용하여 OWL-S[14]로 의미가 기술된 웹 서비스 컴포지션을 수행하는 것이다. 지금까지 연구되고 있는 웹 서비스 컴포지션 시스템의 대부분이 정적으로 기술된 서비스 명세 및 초기 상태에서부터 서비스간의 의미 매칭을 통해 특정한 목표(goal)를 만족시키는 플랜을 생성하는 접근 방법을 채택하고 있다. 그러나 유비쿼터스 컴퓨팅 환경 하에서는 동적인 장치 및 환경 센서 등으로부터 얻을 수 있는 컨텍스트 데이터로부터 추출한 상황 지식과 사용자의 선호도(user preference) 등을 이용하여 능동적으로 서비스를 구성하는 동적 서비스 검색 및 서비스 컴포지션 기술에 대한 연구가 필요하다. 본 장에서는 우선 현재 진행되고 있는 웹 서비스 컴포지션 기술의 기본적인 접근 방법에 대하여 살펴보고 URC를 위한 서비스 컴포지션 기술에 대해서 설명하기로 한다.

1. OWL-S 웹 서비스 온톨로지

자동화된 웹 서비스 컴포지션을 위해서는 기본적으로 각각의 웹 서비스에 대한 의미 기술 방법이 필요하다. OWL을 기반으로 웹 서비스를 기술하기 위해 작성된 OWL-S는 웹 서비스의 의미를 정의하는 상위 온톨로지이다. 즉, 각각의 웹 서비스는 OWL-S 온톨로지의 인스턴스가 되며, 서비스에 이전트는 이러한 서비스 인스턴스에 대한 추론을 통하여 기본적으로 다음과 같은 웹 서비스 태스크를 수행할 수 있다.

- 자동화된 웹 서비스 검색(automatic web service discovery)
- 자동화된 웹 서비스 실행(automatic web service invocation)



(그림 5) Top Level of the Service Ontology

- 자동화된 웹 서비스 컴포지션 및 상호 운용성 (automatic web service composition and interoperation)

OWL-S는 (그림 5)와 같이 웹 서비스를 정의하는 service 온톨로지, 서비스가 무엇을 하는지 (what it does)를 표현하는 service profile 온톨로지, 서비스의 프로세스를 구체적으로 기술하는 (how it works) service model 온톨로지, 마지막으로 서비스의 접근 방법을 기술하는(how to access it) service grounding 온톨로지로 구성된다.

웹 서비스를 기술하기 위한 표준화된 형태로써 이러한 OWL-S의 상위 온톨로지가 사용되며, 각각의 웹 서비스는 OWL-S의 상위 온톨로지에서의 정의된 class와 property를 이용하여 작성된다.

- Service profile

Service profile 상위 온톨로지는 웹 서비스의 구조와 서비스가 수행하는 기능, 그리고 서비스의 특징에 대해 간략히 기술하며, 서비스가 수행하기 위해 필요한 input과 precondition과 수행 후 제공되는 output과 effect를 정의한다. 이를 통해 웹 서비스가 무엇을 하는지를 정의한다.

- Service model

Service model 상위 온톨로지는 process-flow, composition hierarchy, process definition을 정의하며, 각 process의 IOPE를 구체적으로 정의한다. 이를 통해 서비스의 구체적인 프로세스 동작을 기술한다.

• Service grounding

Service grounding 상위 온톨로지에서는 communication protocol, port number 등을 정의하며 웹 서비스가 실제로 수행하기 위해 필요한 모든 정보를 기술한다. 웹 서비스의 실질적인 동작을 처리하며 소프트웨어 에이전트는 이렇게 정의된 프로토콜을 통해 웹 서비스에 접근하고 웹 서비스를 실행시킬 수 있다.

2. HTN 기반 서비스 컴포지션

HTN 기반 서비스 컴포지션의 기본 아이디어는 아주 복잡한 태스크를 하위 태스크(sub task)들로 나누고 이 하위 태스크를 더 나눌 수 없는, 즉 직접 실행이 가능한 가장 간단한 수준의 단위 태스크까지 나누어 가는 task decomposition 방법이다. 이를 위해 HTN은 오퍼레이터와 메소드라 불리는 계층구조를 가지는 태스크 단위로 플래닝을 수행한다. 오퍼레이터는 직접 실행이 가능한 태스크의 최소 단위로서, OWL-S의 atomic process와 매핑된다. 메소드는 하위 태스크로 분할될 수 있는 태스크로서 OWL-S의 컴포지션 프로세스와 매핑된다. OWL-S의 계층구조에서처럼 HTN 역시 메소드를 통한 태스크의 계층구조를 가진다.

HTN은 POP나 STRIPS와는 달리 태스크 네트워크로 goal을 정의한다. HTN의 태스크는 goal task, primitive task, compound task의 3가지 태스크로 정의가 가능하다. 각각의 태스크들은 OWL-S 기술을 기반으로 하는 웹 서비스 컴포지션에서 각각의 process model과 대칭적으로 설명이 가능하다. Goal task는 웹 서비스 컴포지션에서 사용자가 원하는 웹 서비스 컴포지션 목표와 매칭되며, primitive task는 OWL-S process model에서 atomic 프로세스와 매칭된다. 또한 compound task는 OWL-S의 컴포지션 프로세스와 매칭된다. 이렇게 HTN은 OWL-S의 process model과 각각의 태스크들의 매칭이 가능하기 때문에 OWL-S 기반의 웹 서비스 컴포지션 방법

으로써 적합한 것으로 인식되어 일반적인 웹 서비스 컴포지션 기술에 적용되고 있다.

3. Regression 기반 서비스 컴포지션

일반적인 탐색 기반의 서비스 플래닝 방법은 크게 순방향 진행(forward progression) 탐색과 역방향 회귀(backward regression) 탐색 방법으로 나눌 수 있다. 순방향 진행 탐색은 초기상태에서부터 시작하여 주어진 목표상태에 도달하기 위해 요구되는 오퍼레이터들의 순서를 찾는 플래닝 방법이다. 이에 반해 역방향 회귀 탐색은 주어진 목표상

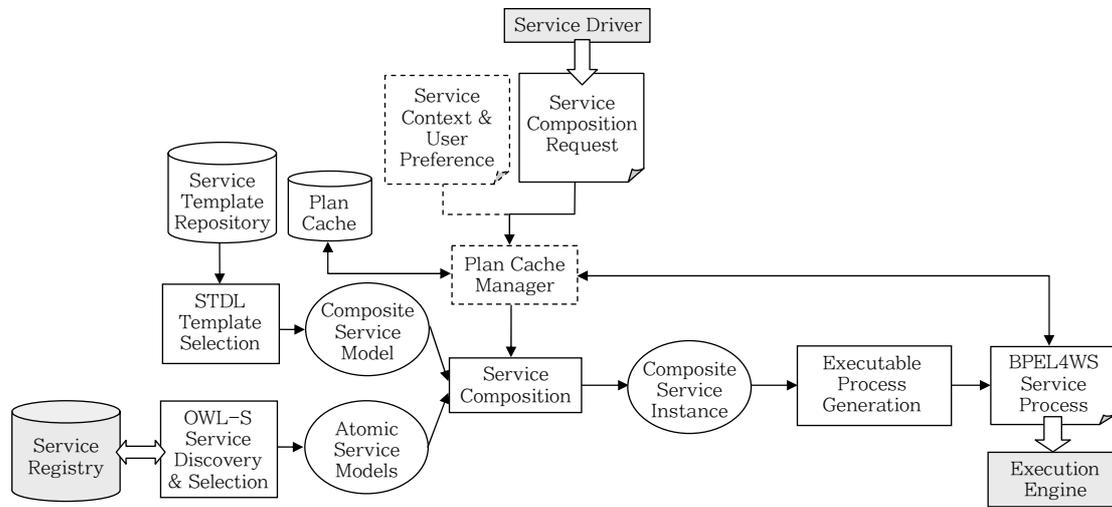
```

Let N = 1
Let B = add_scope_num(G, N)
Let T = {}

Plan(B, T, N):
  if (B is empty)
    then Let P = extract_goal(T)
  else
    Let H = pop(B, B')
    if (H is operator)
      then Let T' = push(T, H)
           Plan(B', T', N)
    else if (H is satisfied by T or I)
      then Let T' = push(T, H)
           Let N' = increment(N)
           Plan(B', T', N')
    else
      Let O = set of operator instantiations that add
        state H(subgoal)
      if O = {}
        then fail and backtracking
      else Let C = conflict_resolution(O)
           Let Prec = precondition states of C
           if (there is no violation with C in T)
             then N' = increment(N)
                  Let Prec' = add_scope_num(Prec, N')
                  Let B' = append(Prec', C, B)
                  Plan(B', T, N')
           else if (there is temporary violation with C in T)
             then reordering(B, T, B', T')
                  N' = increment(N)
                  Let Prec' = add_scope_num(Prec, N')
                  Let B'' = append(Prec', C, B')
                  Plan(B'', T', N')
           else Let V = find_violation_subgoal(C, B, T)
                  regress(B, T, V, B', T)
                  Plan(B', T, N)
  
```

P is a plan.
 S is a finite set of state.
 I is a set of initial states, $I \subseteq S$
 G is a set of (sub)goal states, $G \subseteq S$
 B is a B-stack
 T is a T-stack
 N is a scope number

(그림 6) Goal Regression 기반 서비스 플래닝 알고리즘



(그림 7) URC 서비스 컴포지션

태에서 시작하여 초기상태에 도달하기 위해 요구되는 오퍼레이터들의 역 순서를 찾는 플래닝 방법이다. 이러한 역방향 탐색 기반 플래닝의 경우 순방향 탐색에 비해 플래닝 과정상의 backtracking이 발생할 가능성이 작으므로 웹 서비스 컴포지션에 비교적 적합한 방법으로 연구되고 있다. (그림 6)은 기본적인 goal regression[15] 기반 서비스 플래닝 알고리즘을 나타낸다.

regression과 같은 플래닝 알고리즘을 이용하여 개념적인 서비스 플랜을 생성하게 된다. 생성된 서비스 플랜은 실행 가능한 형태는 아니며, 실제로 검색된 최적의 웹 서비스와의 바인딩을 통해 실행 가능한 BPEL4WS[16] 기반의 서비스 프로세스로 변환된다. (그림 7)은 전체적인 URC 서비스 컴포지션 수행 과정을 보여준다.

4. URC 서비스 컴포지션 기술

URC 서비스 컴포지션 기술은 기존의 컴포지션 방법과 다르게 특정 서비스 도메인에 대한 일반적인 지식을 템플릿으로 정의하고 이러한 서비스 지식을 기반으로 컴포지션을 수행하는 것을 기본적인 메커니즘으로 하고 있다. 이러한 방법의 장점은 템플릿으로 정의된 일반적인 서비스 모델과 실세계에 존재하는 물리적인 서비스 구현을 분리하여 정의함으로써 다양한 서비스가 존재하는 동적인 유비쿼터스 컴퓨팅 환경 하에서 서비스 모델 및 컨텍스트 데이터를 기반으로 각각의 서비스 상황에 최적인 실제 서비스를 자유롭게 선택할 수 있다는 것이다. 서비스 컴포지션을 위한 템플릿은 OWL을 이용하여 정의된 STDL에 의해서 기술되며, STDL로 기술된 컴포지션에 필요한 템플릿을 대상으로 HTN 또는

약어 정리

BPEL4WS	Business Process Execution Language for Web Services
HTN	Hierarchical Task Network
IOPE	Input, Output, Precondition, Effect
OWL	Web Ontology Language
OWL-S	OWL for Services
POP	Partial Ordered Planner
SOM	Self-Organizing Map
STDL	Service Template Description Language
TEA	Technology for Enabling Awareness
URC	Ubiquitous Robotic Companion

참고 문헌

[1] 이석한, 지능로봇기술: 연구 및 산업화 동향, 전자공학

- 회지 제32권 1호, 2005, pp.14-25.
- [2] 김상훈, 이영직, 신성장동력산업의 기반, 음성인터페이스 기술, ETRI CEO Information 12호, 2004.
- [3] Jose Mira et al., The Knowledge Engineering Approach to Autonomous Robotics, IWANN 2003, LNCS 2687, 2003, pp.161-168.
- [4] 이기중, "정보추출(Information Extraction)," <http://nlp.korea.ac.kr/~kjlee/ie.html>
- [5] Anand Ranganathan and Roy H. Campbell, "A Middleware for Context-Aware Agents in Ubiquitous Computing Environments," *In ACM/IFIP/USENIX Int'l Middleware Conf.* 2004, Rio de Janeiro, Brazil, June 16-20, 2004.
- [6] Anind K. Dey, "Understanding and Using Context," *Personal and Ubiquitous Computing, Special Issue on Situated Interaction and Ubiquitous Computing*, Vol.5, No.1, 2001.
- [7] M. Korkeaaho, "Context-Aware Applications Survey," <http://www.hut.fi/~mkorkea/doc/context-aware.html>
- [8] M. Roman et al., "Gaia: A Middleware Infrastructure to Enable Active Spaces," *In IEEE Pervasive Computing*, Oct.-Dec. 2002, pp.74-83.
- [9] N.Q. Hung et al., "A Survey on Middleware for Context-Awareness in Ubiquitous Computing Environments," *Korea Information Society Review*, Vol.10.4, July 2003. pp.97-121.
- [10] Hans-Werner Gellersen, Albrecht Schmidt, Michael Beigl: Multi-Sensor Context-Awareness in Mobile Devices and Smart Artifacts, *Mobile Networks and Applications*, Vol.7, No.5, Oct. 2002, pp.341-351.
- [11] W3C, Web Service Description Working Group, <http://www.w3.org/2002/ws/desc/>, 2002.
- [12] D.S. Nau, T.C. Au, O. Ilghami, U. Kuter, J.W. Murdock, D. Wu, and F. Yaman, "SHOP2: An HTN Planning System," *J. of Artificial Intelligence Research*, Vol.20, AIAA, 2003, pp.379-404.
- [13] K. Erol, D. Nau, and J. Hendler, "UMCP: A Sound and Complete Planning Procedure for Hierarchical Task-Network Planning," *AIPS-94*, Chicago, 1994.
- [14] OWL-S Home Page, "OWL-S: Semantic Markup for Web Services," <http://www.daml.org/services/owl-s/1.1/overview/> 2004.
- [15] J.L. Pollock, "The Logical Foundations of Goal-Regression Planning in Autonomous Agents," *Artificial Intelligence*, 1998, pp.267-335.
- [16] T. Andrews et al., "BPEL for Web Services," 2003, <http://www.ibm.com/developerworks/library/ws-bpel/>