

다중 에이전트 기반의 자율적 계획수립과 계획실행

Multiagent-based Autonomic Planning and Plan Execution

이선아, 황경순, 이건명

충북대학교 전기전자컴퓨터공학부, 첨단정보기술연구센터(AITrc)
kmlee@cbnu.ac.kr

요 약

자율컴퓨팅은 관리자나 사용자의 개입없이 시스템이나 서비스가 원활하게 동작되는 환경으로 구조조정, 최적화, 치료, 방어 등을 위한 여러가지 기술이 복합적으로 요구된다. 태스크를 자율적으로 완수하고 최적으로 수행하기위한 자율적 계획수립 및 계획실행은 자율컴퓨팅에서 필요한 요소기술이다. 이 논문에서는 목표의 실패없는 수행을 위한 여러 잠재적인 실행경로를 포함하는 강건한 계획수립과 다중 에이전트 구조를 이용하여 가용한 자원에 대한 전반적인 고려와 함께 실행시점의 상황을 반영하여 최적의 실행경로에 따라 계획을 실행하게 하는 방법을 제안한다.

Key Words : 계획수립, 적응적 계획실행, 자율컴퓨팅, 다중 에이전트

1. 서 론

서비스 제공을 위한 유지비용의 절감 및 최신, 최적의 서비스 보장을 위해서 아웃소싱을 통해서 많은 일을 수행해가는 추세이다. 비즈니스 현장에서는 직접대응생산(agile manufacturing), 가상기업(virtual enterprise), 간판(kanban) 기반의 JIT(just in time) 시스템 등이 대표적인 예이고, 정보서비스 분야에서는 분산객체를 사용하여 서비스를 조합하여 제공하는 웹서비스가 대표적이다. 이들의 대표적인 특징은 서비스 구현을 위한 요소들이 자체적으로 보유하고 관리하지 않고, 제3자를 통해서 이를 구현하는 것이다. 비즈니스 현장의 유동성과 서비스 제공자의 경쟁 및 서비스 제공 제약 등을 고려하여 서비스 구현을 위한 서비스 요소를 제공하는 제공자를 동적으로 변경하는 것이 바람직한 상황도 있다. 한편, 특정 서비스 태스크를 구현하는 경우, 서비스를 구현하는 여러 가지 방법이 있을 수 있으므로, 상황에 따라 최적의 서비스 요소를 사용하여 서비스 태스크를 구현하는 것이 바람직하다.

정보시스템 분야에서는 사용자의 개입없이 시스템이나 서비스가 자율적으로 운영될 수 있도록 하는 자율 컴퓨팅(autonomic computing)에 대한 관심이 커지고 있다.[7] 자율 컴퓨팅 환경은 서비스 구성요소를 자율적으로 구성하는 자가 구성(self configuration), 시스템이나 서비스에 문제가 있을 때 자율적으로 고치는 자가 치료(self healing), 최적으로 시스템이나 서비스를 운영하도록 하는 자가

최적화(self optimization), 외부의 공격이나 침해에 대해 자율적으로 대응할 수 있는 자가 방어(self protection) 기능을 제공할 수 있는 것을 말한다. 자율 컴퓨팅의 모든 속성을 갖춘 시스템이나 서비스를 구현하는 것은 아직 많은 기술의 발전을 필요로 하지만, 자율 컴퓨팅의 일부 기능의 채택을 통해 시스템이나 서비스를 구현하게 되면, 많은 장점을 제공할 수 있게 된다.

이 논문에서는 자가 구성과 자가 최적화를 통해 외부의 서비스를 활용해서 서비스를 제공하는 다중 에이전트 기반 환경을 소개한다. 제안하는 방법은 일반 비즈니스 환경에서도 적용될 수 있지만, 기본적으로 분산 서비스 객체를 이용하여 정보 서비스를 구성하여 제공하는 것을 대상으로 한다. 제공되는 서비스가 단발성으로 요청되는 것이 아니라 반복적으로 발생하는 서비스인 것을 대상으로 한다. 또한 서비스를 제공하기 객체 또는 요소가 복수개의 공급자에 의해서 제공되고, 공급자의 상황에 따라 서비스의 가용여부, 서비스 제공 비용, 서비스 가용 시간 등이 동적으로 변할 수 있다는 것을 가정한다. 또한 서비스가 서비스 객체 또는 요소의 한 가지 절차 또는 구성으로만 구현되는 것이 아니라 여러 가지 방법으로 제공될 수 있는 융통성있는 환경인 것을 가정한다. 한편, 서비스를 제공할 수 있는 새로운 서비스 객체나 요소, 방법이 동적으로 추가될 수도 있기 때문에, 기존에는 고려되지 않은 새로운 방법으로 서비스가 구현될 수 있다. 웹서비스와 같은 경우는, 특정 서비스 요소가 복수 개의 서비스 객체에 의해서 제공될 수 있으며, 이들의 서비스 가용 및 비용에 대한 가변성이 있으므로, 이 논문에서 제시하는 조건에 부합하는 것이다. 물론 비

본 연구는 첨단정보기술연구센터(AITrc)를 통해서 과학재단 지원으로 수행된 것임.

즈니스 현장에서도 이와 같이 조건이 만족되는 것을 쉽게 찾을 수 있다.

이와 같은 상황에서 어떻게 하면 효과적으로 서비스를 자율적으로 구성하여, 자율적으로 최적화하면서 서비스를 제공할 것인가가 이 논문에서 해결하고자 하는 것이다. 이러한 동적인 환경의 서비스 제공을 위해서는 주어진 서비스 태스크를 실현할 수 있는 작업 방법에 대한 작업계획(planning)이 이루어져야 하고, 이 작업계획을 참조하여 최적으로 서비스를 실현해야 한다. 작업계획의 실행은 서비스 요소 또는 객체의 가용여부, 비용 등의 동적인 상황을 반영해야 하기 때문에, 이를 반영해서 이루어져야 하고, 경우에 따라서는 작업계획을 새로 해야 하는 상황이 발생할 수 있다. 이 논문에서는 강건한 계획(robust plan)을 만들고, 다중 에이전트 구조를 이용하여 동적인 상황을 반영하여, 최적으로 계획을 실행하는 방법을 제안한다.

2절에서는 관련연구로서 다중 에이전트 시스템 관련 기술, 강건한 계획수립 방법 등에 대해서 소개하고, 3절에서는 제안한 다중 에이전트 기반 자율적 계획 수립 및 실행기법에 대해서 소개하고, 4절에서 결론을 맺는다.

2. 관련연구

다중 에이전트 시스템은 위임받은 일을 자율적(autonomous)으로 수행하는 소프트웨어나 시스템 모듈인 에이전트들이 상호작용하여 특정 목적이나 목표를 위해서 일을 하는 조정된(coordinated) 환경의 시스템이다. 다중 에이전트 시스템 분야에서는 에이전트의 구조, 통신 및 언어, 작업 조정 및 협력, 디렉토리 구조, 개발 운영 프레임워크 등이 많이 연구되어 왔다. 에이전트 구조로는 단순 if-then 형태로 기능이 기술되고 동작하는 반응(reactive) 에이전트 구조, 사실정보인 Belief, 직접적인 욕구에 대응하는 Desire, 결심한 목표인 Intention을 고려하여 숙고적으로(deliberate) 일을 수행하도록 하는 BDI 구조 등 여러 구조가 연구되어 왔다.[2,3,5] 에이전트 통신 및 언어로서 ACL, KQML 등의 언어 및 speech act 이론 등이 연구 개발되었다. 작업 조정 및 협력 연구에서는 효과적인 작업수행을 위한 협력(collaboration), 경쟁해소, 작업 위임 등의 작업 조정과 팀구성, 신뢰구축 등에 대한 것이 관심있게 다루어져 왔다. 디렉토리에 대한 연구에서는 각 에이전트의 기능 및 역할, 접근 및 사용방법을 효과적으로 제공하기 위한 방법이 다루어져 왔다. 다중 에이전트 개발 운영환경으로는 Java 환경에서 제공되는 프레임워크 등이 연구되어 왔으며, 웹서비스 환경을 활용하여 에이전트를 개발하는 환경에 대한 연구도 수행되어 왔다.[4]

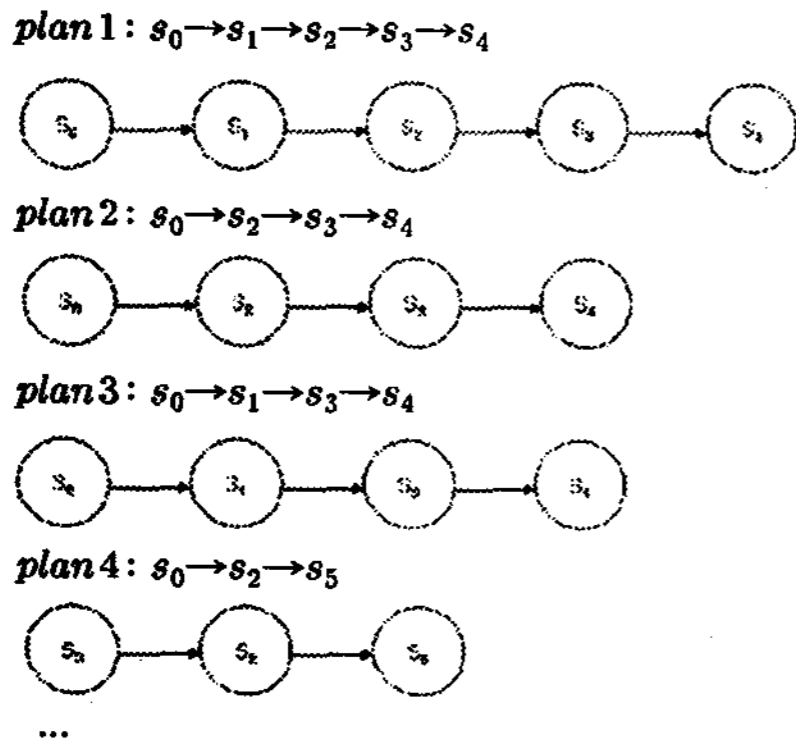
작업의 위임 및 서비스 제공의 선정을 위한 방법으로 contract-net protocol, 디렉토리 기반 방법 등이 사용될 수 있다. contract-net protocol에서는 서비스 제공자를 선정하기 위해 우선 작업을 위임할 에이전트가 해당 작업을 관련 에이전트에게 공고(announce)를 한다. 이를 받은 에이전트들이 자신이 해당 작업을 할 수 있으면, 자신의 작업실행 조건과 함께 입찰(bid)을 하게 된다. 작업 위임자는 응찰할

에이전트 중에서 최적인 것을 선택하여 해당 작업을 위임(award)하게 된다. 디렉토리 기반 기법에서는 모든 서비스 제공자들이 자신이 기능 및 역할에 대한 정보 및 접근방법에 대한 정보를 디렉토리에 에이전트에 등록하여 공개(publish)하면, 작업을 위임할 에이전트가 디렉토리에 정보를 요청하여, 작업을 위임할 에이전트를 결정하여, 서비스를 요청하게 된다. 웹서비스 환경에서 이러한 역할을 수행하는 디렉토리로서 UDDI가 표준화되어 구현되어 진화하고 있다. 또한 경매(auction) 메커니즘을 이용하여 서비스 제공자를 결정하는 방법도 고려되고 있다.

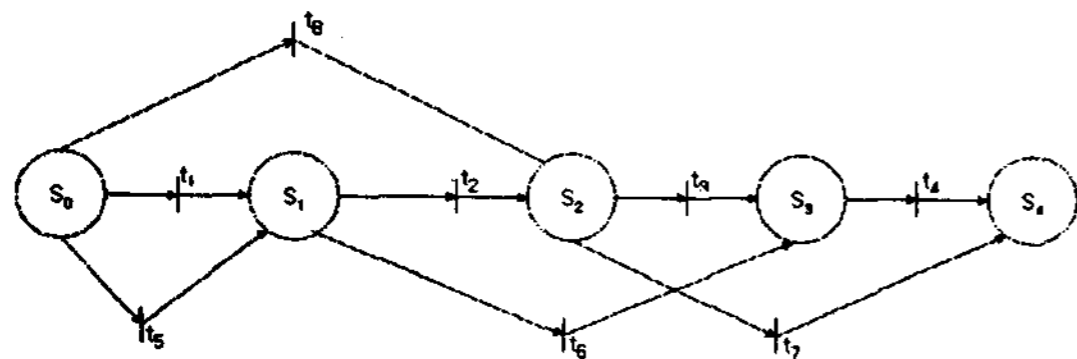
단순한 작업이 아닌 복잡하고 쉽지 않은 작업인 경우에는 하나의 에이전트나 서비스 제공자에 의해서 수행될 수 없고, 여러 에이전트나 서비스 제공자가 참여해야 한다. 또한 이러한 작업을 수행하는 방법이 미리 정해질 수 없는 상황도 있고, 새로운 에이전트나 서비스 제공자가 참여할 수 있기 때문에, 작성 수행 방법을 미리 정해 둘 수 없는 경우가 생길 수 있다. 이러한 상황에 대처하기 위해 작업계획을 동적으로 수립하는 방법이 사용된다. 계획수립은 그것을 실행하는데 시간이나 공간에 관련된 많은 자원들을 요구하는 매우 어려운 문제이다. 대표적인 계획수립 접근방법으로는 상태-공간 계획수립, 계획-공간 계획수립, 그리고 HTN 계획수립 방법 등이 있다.[6] 상태-공간 계획수립과 계획-공간 계획수립에서는 계획수립 도메인을 단위 작업의 수행전 상태와 수행후 상태를 포함하는 연산자(operator)들로 기술한다. 상태-공간 계획수립 방법에서는 초기상태를 목표상태로 만들기 위해 연산자를 선택해 가는 방법으로 계획을 수립한다. 계획-공간 계획수립 방법에서는 목표상태를 결과로 갖는 부분계획에서 출발하여, 해당 부분계획의 전제조건을 만족시키는 부분계획을 찾아가는 것을 반복하는 방법 등으로 계획을 수립한다. HTN 계획수립에서는 계획수립 도메인을 연산자와 메소드(method), 공리(axiom) 등으로 기술하고, 계획인 메소드를 이용하여 재귀적으로 주어진 태스크를 분할하여, 최종적으로 연산자들로만 구성된 계획을 만들어 낸다.

특정 작업의 수행을 위해 계획을 수립하여 이를 실행하고 있는 중에 환경의 변화로 더 이상 계획을 진행할 수 없는 경우에는, 새로이 계획을 다시 세워야 한다. 이러한 재계획수립(re-planning)은 시간적, 자원적 비용이 크기 때문에, 계획 실행중에 가능하면 피하는 것이 바람직하다. 이를 위해 저자들의 기존 연구[1]에서 개발한 강건한 계획수립 방법을 사용하는 것을 제안하였다. 이 방법에서는 목표 달성을 위한 모두 가능한 계획을 찾아내고, 이를 모두 포함하는 Petri net 기반의 방법으로 강건한 계획을 만들어 낸다. 그림 1은 강건한 계획수립의 예를 보인 것으로 그림 1-(a)는 가능한 계획들을 나타낸 것이고, 그림 1-(b)는 이들 가능 계획을 모두 포함하는 Petri net 형태로 표현된 강건한 계획을 보인 것이다. 강건한 계획을 실제 해당 태스크의 수행을 할 때는 현재 태스크를 수행할 에이전트 또는 서비스 객체의 가용상태를 확인하여 해당 태스크를 수행할 수 있는 최적의 에이전트를 이용한다. 더 이상 계획을 진행할 수 없는 상황에서는 Petri net상에서 다른 실행경로를 따라갈 수 있는 이전 단계로 돌아가

서 다른 계획 실행경로를 따라 가게 된다.



(a) 가능한 계획들의 예



(b) Petri net으로 표현된 강건한 계획

그림 1. Petri net을 통한 강건한 계획 표현[1]

3. 자율적 계획 수립 및 계획 실행

3.1 문제 정의

서비스 객체 또는 요소를 외부에서 지원받아 서비스를 구성하여 제공하는 환경으로, 특성 서비스가 단발성으로 발생하는 것이 아니라, 반복적으로 발생하는 것들을 대상으로 한다. 외부의 서비스 객체 또는 요소는 가용여부, 비용, 위임받은 일에 대한 성공적 완료가 동적으로 변할 수 있는 상황이다. 한편, 서비스 객체 및 요소는 비즈니스나 서비스 진화 과정에서 새로 만들어지거나 소멸될 수 있으며, 이와 함께 서비스 실현방법도 새로이 만들어질 수 있다. 이와 같은 동적인 환경에서의 서비스 제공을 위해서 서비스 객체 또는 요소를 미리 구성해 두는 것이 아니라, 필요할 때 동적으로 만들어 사용하는 것을 전제로 한다. 또한 서비스 객체를 사용할 때는 최적으로 서비스가 제공될 수 있도록 하는 것을 목표로 한다.

이와 같은 서비스 동적 구성 및 최적화를 위해서 본 논문에서는 자율컴퓨팅의 관점에서 필요에 따라 서비스를 자율적으로 재구성하고, 서비스 객체 또는 요소를 자율적으로 최적으로 선정하여 사용하는 다중 에이전트 방법을 제안한다.

3.2 다중 에이전트 시스템 구성

자율적 서비스 구성 및 최적화를 여러 역할을 나누어서 담당하는 다중 에이전트 시스템 구조를 이용한다. 기본적으로 모든 서비스 객체나 서비스 요소는 개별 에이전트로 동작하며, 서비스 제공을 위한 정보공개, 서비스 제공 의사 표명 및 해당 시점

서비스 가용여부 및 비용 정보제공과 관련한 정보 교환, 위임된 태스크에 대한 수행 및 결과 전달 등을 담당한다. 서비스 구성 에이전트는 특성 서비스 제공을 위해, 서비스 객체나 서비스 요소들로 조합된 계획수립을 관리하고, 구성된 계획에 따라 서비스 객체 에이전트나 서비스 요소 에이전트에게 작업을 위임하여 수행하도록 하는 실행 조정을 하는 역할을 한다. 디렉토리 에이전트는 서비스 객체 에이전트나 서비스 요소 에이전트가 자신의 기능 및 역할, 접근방법 등을 등록할 수 있도록 하고, 서비스 구성 에이전트에게 등록된 서비스 객체나 요소를 검색할 수 있도록 하는 역할을 한다. 서비스 계획 수립 에이전트는 서비스 구성에 관련된 정보 즉, 계획구성을 위한 메소드, 연산자 정보를 유지관리하고, 서비스 구성 에이전트로부터 서비스 제공을 위한 계획 수립 요청을 받아, 서비스 제공을 위한 계획을 생성하여 제공하는 역할을 한다.

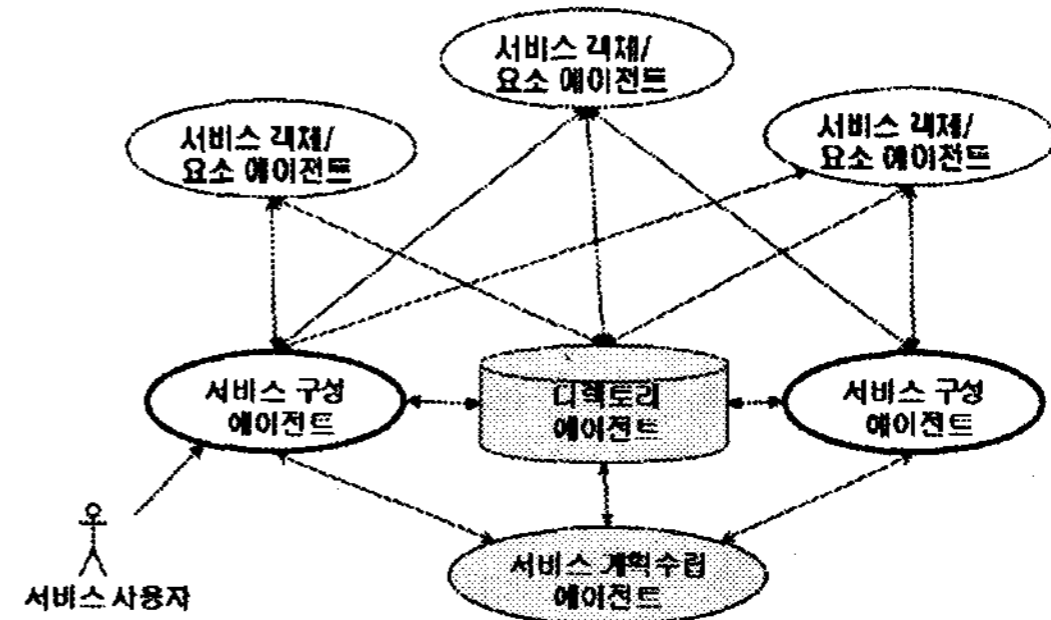


그림 2. 자율적 계획수립 계획실행 다중 에이전트 구성

3.3 서비스 제공 계획 수립

서비스 사용자로부터 서비스구성 에이전트가 어떤 서비스에 대한 요청을 받으면, 해당 서비스를 구성하기 위해 서비스에 대한 요구사항을 서비스 계획수립 에이전트에 전달한다. 서비스 계획수립 에이전트는 받은 요구사항과 자신이 보유한 계획수립 지식을 이용하여 서비스 구성을 위한 작업계획을 수립한다. 계획수립 에이전트는 HTN 기반 계획수립 기법을 구현한 JSHOP2[8]를 이용하여 저자 등 [1]이 제안한 Petri net으로 표현된 강건한 계획을 생성하여, 서비스 구성 에이전트에게 전달한다. 서비스 계획수립 에이전트는 디렉토리 에이전트와 동기화되어 특정 서비스 객체 또는 요소 에이전트가 다중 에이전트 환경에서 삭제되어 역할을 할 수 없는 경우, 이 에이전트와 관련된 메소드나 연산자를 삭제하는 등의 관리를 통해, 서비스 환경의 상황을 동적으로 관리하도록 한다. 이와 같은 동적 계획수립 지식의 관리를 통해, 동적으로 변화는 상황에 적응적인 계획수립을 통해, 자율적인 서비스 구성의 한 역할을 한다.

3.4 자율적 서비스 계획 실행

서비스 계획수립 에이전트로부터 Petri net 형태의 강건한 계획을 받으면, 서비스 구성 에이전트는

이 계획에 따라 서비스 객체 및 요소 에이전트의 작업 수행을 통해서 기대하는 서비스를 사용자에게 전달하게 된다. 강건한 계획은 서비스를 실현을 위한 모든 가능한 계획을 포함하고 있으며, 각 서비스 구성요소를 어떤 서비스 객체 에이전트나 서비스 요소 에이전트가 담당할지는 미정인 상태이다. 따라서 실제 서비스 계획을 실행할 때는 최적의 서비스 경로 선택과 함께 서비스를 담당할 에이전트를 선택해야 한다. 서비스 객체 에이전트나 서비스 요소 에이전트의 서비스 가용여부, 서비스 비용 등이 동적으로 변하는 환경이기 때문에, 서비스 구성 에이전트는 이러한 상황을 반영하여 계획 실행경로를 선정하고, 실제 서비스 구성요소를 담당할 에이전트를 결정해야 한다.

서비스 구성 에이전트는 주어진 강건한 계획을 참고하고, 각 구성요소를 담당할 수 있는 서비스 객체/요소 에이전트에 대한 정보를 디렉토리 에이전트로부터 입수한다. 입수된 에이전트 목록을 참고하여, 각 구성요소를 수행할 수 있는지 여부 및 비용 등에 대한 정보를 이들 서비스 객체/요소 에이전트에게 contract-net protocol과 유사한 방법으로 공고-입찰의 형태로 연결관계를 설정한다. 각 구성요소별 최적 서비스 비용을 결정하여, Petri net 으로 표현된 강건한 계획을 구성하는 각 서비스 요소(트랜지션에 해당)별 최소 잔여 비용을 최종단 요소로부터 역방향으로 계산한다. 이 방법은 Dijkstra 알고리즘[12]을 마지막 상태에 해당 플레이스(place)를 시작 노드로 하여 적용하는 방식으로 구현한다.

강건한 계획을 표현한 Petri net에서 최소 잔여 비용이 계산되면, 계획을 실행하는 매 단계에서, 최소 잔여비용을 주는 실행 경로를 선택하고, 해당 요소를 실행할 에이전트로서, 해당 요소에 대응하는 등록된 에이전트에게 작업을 의뢰한다. 작업을 의뢰 받은 에이전트가 어떤 이유에서 작업을 완수할 수 없는 상황이 발생한 경우, 대안의 에이전트가 있는 경우에는 대안의 에이전트에게 해당 작업요소를 다시 위임할 수 있도록 서비스 구성 에이전트가 지속적으로 작업을 위임한 에이전트를 관찰하고 조정하는 역할을 담당한다. 한편, 대안의 에이전트가 없는 경우에는 Petri net의 해당 노드에서 역방향으로 돌아가면서, 실행가능한 최적의 비용을 주는 작업요소 노드까지 돌아간다. 이를 위해 어떤 서비스가 완료되지 전에는 중간 결과를 임시로 보관하도록 한다. 한편, 계획의 실행중에 작업계획에 연결되어 서비스 관련 정보를 공유하고 있는 에이전트에서 서비스 품질 관련 정보가 바뀐 경우, 이 정보가 반영되어 최소 잔여 비용이 수정되어, 진행되는 작업계획에서 현 상황을 반영한 최적의 작업 실행이 될 수 있게 한다. 한편, 새로운 서비스 객체 또는 요소 에이전트가 새로 추가되는 경우에는 이 정보가 디렉토리 에이전트에 등록되기 때문에, 서비스 구성 에이전트는 주기적으로 디렉토리 에이전트를 참고하여, 새로운 에이전트와 관계를 설정하여, 작업계획 실행에 반영하도록 한다.

현재 주어진 강건한 계획에서 서비스 실현이 더 이상 불가능하다고 판단되면, 서비스 계획수립 에이전트에게 다시 작업계획을 생성할 것을 요구하고, 새로운 작업계획이 기존 것과 다른 실행 경로를 포

함하고 있으면, 이 실행경로를 실행하고, 그렇지 않으면 서비스 실패를 사용자에게 알린다.

4. 결론 및 향후과제

동적으로 변화하는 환경에서 외부의 서비스 객체들이나 서비스 요소들을 사용하여 서비스 실현하고자 하는 다양한 시스템에서, 사용자의 주도적인 개입없이 자율적으로 작업계획을 생성하여 효과적으로 최적의 서비스 제공하는 것은 핵심적인 기술이 될 것이다. 이 논문에서는 이러한 환경에 적용할 수 있는 강건한 작업계획을 수립하고, 상황에 따라 자율적으로 자원을 활용하는 다중 에이전트 기반의 방법을 제안하였다. 자율 컴퓨팅에서 기대되는 자기방어와 자가 치료 등의 특성을 포함하는 확장 발전시켜 보다 자율적인 시스템을 구성하는 방법에 대한 연구가 진행 중이다.

참 고 문 헌

- [1] 황경순, 이승희, 이진명, "지능적인 웹서비스를 위한 강건한 계획 생성과 동적 실행 방법", 한국퍼지및지능시스템학회 2007춘계학술대회 논문집, 제17권 제1호, pp.320-323, 2007.04.
- [2] 손봉기, 김학준, 이진명, "계획생성과 실행을 위한 규칙 기반 에이전트 구조", 2003년도 한국정보과학회 춘청지부 추계 학술발표 논문집, 제15권, 1호, pp.179-182, 2003.
- [3] 손봉기, 김학준, 이진명, "목표 지향 및 사건 지향 작업 처리를 위한 규칙 기반 에이전트 추론", 2003년도 한국정보과학회 춘청지부 추계 학술발표 논문집, 제15권, 1호, pp.183-186, 2003.
- [4] 황경순, 이승희, 양원섭, 이진명, "웹서비스 기반의 개방형 다중 에이전트 시스템 구조", 한국퍼지및지능시스템학회 2006추계학술대회 논문집, 제16권 제2호, pp.3-6, 2006.11.
- [5] 손봉기, 이진명, "규칙기반 BDI 에이전트 구조", 한국퍼지및지능시스템학회 2004춘계학술대회 학술발표논문집, 제14권1호, pp.75-78, 2004. 10.
- [6] M. Challab, D. Nau, P. Traverso. *Automated Planning : Theory and Practice*. Morgann Kaufmann.(2004)
- [7] P. K. McKinley, S. M. Sadjada, E.O. Kasten, B.H.C. Cheng, *Composing Adaptive Software*, IEEE Computer, pp.56-64, July 2004.
- [8] D. Nau, Y. Cao, A. Lotem, and H. Munoz-Avila. SHOP : Simple Hierarchical Ordered Planner. *Proc of IJCAT-99*, pp.968-973, 1999.
- [9] A. Blum, M. Furst. Fast Planning Through Planning Graph Analysis, *Artificial Intelligence*. 90. pp.281-300, 1997.
- [10] N. Gibbins, S. Harris, N.Shadbolt. Agent-based Semantic Web Service. *J. of Web Semantics*. 1. pp.141-154, 2004.
- [11] E. Sirin, B. Parsia. Planning for Semantic Web Service. *Proc of the 3rd International Semantic Web Conference*. 2004.
- [12] T. Cormen, C. Leiserson, R. Rivest, C. Stein, *Introduction to Algorithms*, McGraw Hill, 2001.