

# 상황인지 기반 최적화가 가능한 개인화된 모바일 웹서비스 구축을 위한 다중에이전트 접근법에 관한 연구\*

권오병\*\* · 이주철\*\*\*

## A Multi-Agent Approach to Context-Aware Optimization for Personalized Mobile Web Service\*

Oh-byung Kwon\*\* · Ju-chul Lee\*\*\*

### ■ Abstract ■

Recently the usage of mobile devices which enable the accessibility to Internet has been dramatically increased. Most of the mobile services, however, so far tend to be simple such as infotainment service. In order to fully taking advantage of wireless network and corresponding technology, personalized web service based on user's context could be needed. Meanwhile, optimization techniques have been vitally incorporated for optimizing the development and administration of electronic commerce. However, applying context-aware optimization mechanism to personalized mobile services is still very few. Hence, the purpose of this paper is to propose a methodology to incorporate optimization techniques into personalization services. Multi agent-based web service approach is considered to realize the methodology. To show the feasibility of the methodology proposed in this paper, a prototype system, CAMA-myOpt(Context-Aware Multi-Agent system for my Optimization), was implemented and adopted in mobile comparative shopping.

Keyword : Context-Aware Computing, Context-Aware Optimization, Ontology, Web Service, Mobile Service, Multi-Agent Technology

논문접수일 : 2004년 7월 27일      논문게재확정일 : 2004년 10월 15일

\* 본 연구는 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 유비쿼터스 컴퓨팅 및 네트워크 원천기반 기술개발사업의 지원에 의한 것임.

\*\* 경희대학교 국제경영학부

\*\*\* 한동대학교 대학원 경영경제학과

## 1. 서 론

모바일 상거래 서비스는 이동성(mobility)과 휴대성(portability)이 증강된 전자 상거래 서비스이다. 즉 기존 전자상거래 서비스와는 달리 사용자가 이동중에 있는 상황에서도 서비스가 가능하므로 서비스의 기민성과 현장성을 신장시킬 수 있는 장점이 있다. 또한 휴대용 단말기나 PDA와 같은 모바일 상거래의 클라이언트용 장비는 고도로 특정 개인에 소유되는 장비로서 다른 장비에 비해 개인 점유도가 상대적으로 높은 특징이 있다. 이러한 장점과 특징들은 모바일 상거래 서비스가 일반 전자상거래 서비스보다 개인화에 대한 요구가 더욱 높을 것을 예상케 한다.

개인화란 고객의 선호도를 학습하고 그에 맞게 재화를 선정할 수 있으며, 서비스의 질을 향상시키기 위해 선정 결과를 평가할 수 있는 능력을 말한다[14]. 개인화 서비스의 질을 높이기 위한 방도로써 다음과 같은 세 가지의 방식이 시도되고 있다. 첫번째는 개인의 선호도를 활용한 차별화된 서비스이다. 이는 사용자의 프로파일 정보와 선호도 관련 정보를 사용자로부터 입력 받아 특정의 위치에 저장한 다음에 서비스할 내용을 선택적으로 인식하고 편집하여 수동 혹은 자동으로 사용자에게 제공하는 방식을 보편적으로 따르고 있다. 웹상에 존재하는 최근의 개인화 웹사이트는 대체로 이러한 접근 방법을 채택하고 있다. 둘째는, 서비스의 자동화이다. 서비스 시작 시점이나 조건 등을 사용자의 그때마다의 요구에 의하지 않고 스스로의 판단에 의해 결정한 다음 사용자가 위임한 범위 내에서 자율적으로 서비스를 구동하는 경우이며, 에이전트 기법을 주로 활용하여 구현하고 있다. 셋째로, 최근에는 사용자의 현재 상황(context)을 고려한 상황인지 서비스가 등장하고 있다. 사용자의 현재 위치, 사용하고 있는 장비, 그리고 일정 등 매우 다양한 상황 중에서 자동으로 인지 가능한 것들을 센서로부터 입력받아서 그에 따라 개인화된 서비스를 제공하는 것을 지향하고 있다. 이중 가장 활발

한 것은 사용자 위치기반 서비스(LBS, Location-Based Services)이다. 이러한 세가지의 개인화 서비스 기법들을 통합적으로 적용하면 사용자의 개인 특성이나 과업 특성 등 다른 조건이 동일할 경우 서비스의 질을 더욱 올릴 수 있을 것으로 보인다. 그런데 이러한 개인화 서비스를 위한 세가지 방법을 통합적으로 제공하고 있는 서비스는 아직 존재하지 않고 있으며, 프로토타입 시스템 역시 거의 소개되지 않고 있다.

한편, 전자상거래에 경영과학의 최적화 기법을 적용할 경우의 가능성에 대해서는 지속적으로 제기되고 있는 바이며, 그 적용 영역은 온라인 경매 뿐 아니라 동적인 가격 설정, 수익 관리, 최적화 기법을 활용한 데이터 마이닝, 가상 주식 시장 등 매우 다양하다[10]. 전자상거래에 있어서 최적화 기법을 도입하는 것은 시스템의 최적화와 서비스의 최적화로 나뉘어진다. 시스템의 최적화는 서비스 네트워크 모델의 최적화 등을 의미하며 관련 연구가 많이 진행되어 있다. 그러나 서비스 최적화 입장에서 구체적으로 최적화 기법을 도입하여 실현 가능성을 보인 연구는 매우 드물며, 아직은 알고리즘 차원에서 선호도 학습을 위해 최적화 기법을 사용하는 연구 정도가 진행된 상태이다[14].

이에 비해 본 연구의 관심은 모바일상거래에서의 개인화 서비스를 서비스 최적화 입장에서 어떻게 구축할 것인지에 대해 집중하고 있다. 또한 앞서 언급한 세가지 개인화 서비스의 대안을 통합하는 구조를 제공하려는 연구 동기를 가지고 시작했다.

따라서 본 연구의 목적은 개인의 선호도를 고려하고 상황인지가 가능한 다중 에이전트 시스템을 기반으로 하여 개인화 서비스가 가능한 모바일 웹 서비스를 제안하고 실현 가능성을 보이는 것이다. 특히 여러 가지의 개인화 서비스 중에서 최적화 기법을 활용할 수 있도록 최적화모형을 저장하고, 상황에 맞는 선정(selection) 및 수립(formulation)을 지원하는 방법론을 구축하는 것을 주된 연구 대상으로 한다. 또한 에이전트와 웹서비스 사이의 자동

화된 의사소통을 위해 온톨로지를 활용하였다. 즉, 사용자 상황 온톨로지, 사용자 온톨로지, 서비스 온톨로지 등을 구축하여 사용자에게 의해 입력될 정보의 양을 최소화하도록 하였다.

제안하는 시스템의 실현 가능성을 입증하기 위해 다중 지능형 에이전트에 입각한 프로토타입인 CAMA-myOpt(Context-Aware Multi-Agent System for myOptimization)을 구축하였으며, 이를 비교구매투라고 하는 모바일 서비스에 적용하였다.

본 논문의 구성은 다음과 같다. 2장에서는 본 연구와 관련된 상황인지 기반의 모바일 서비스 최적화의 개념과 구조에 대하여 다루고, 3장에서는 구현된 상황인지 모바일 서비스의 프레임워크와 그 내부 요소들에 대해 기술하였다. 4장에서는 3장에서 설명된 프레임워크를 토대로 직접 구현한 결과를 보여주고 그에 대한 설명을 기술하였다. 끝으로 5장에서는 모바일 서비스 분야에 대한 본 연구의 시사점과 앞으로 계속적으로 보완되어야 할 부분에 대해 기술함으로써 결론을 내었다.

## 2. 상황인지 기반의 모바일 서비스 최적화

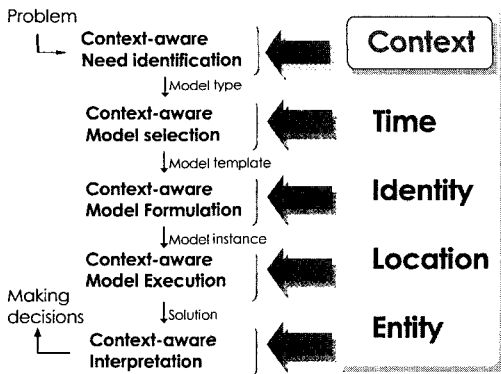
전자상거래에 있어서 최적화 기법을 도입하는 것이 시스템의 최적화와 서비스의 최적화로 나누어지는데, 전술한 대로 본 연구의 관심은 서비스 최적화이다. 특별히 모바일 서비스를 염두에 두고 상황인지 기능이 부가되었을 경우의 서비스 최적화를 대상으로 한다. 서비스의 최적화를 염두에 둘 때 가장 핵심적인 것은 최적화 모형화(optimization modeling)이다. 일반적으로 모형화는 모형화에 대한 요구의 인식(need identification), 모형의 선택(model selection), 모형의 수립(model formulation), 모형의 실행(model execution), 도출된 해의 해석(interpretation)과 같은 단계로 이루어진다. 이 중에서 상황인지 능력에 의하여 지원될 수 있는 부분은 <그림 1>과 같으며 다음과 같이 요약하여 설명된다.

- 상황인지적 모형화 인식 : 최적화를 수행하여 의사결정을 해야 할 문제가 발생했는지를 상황 정보로부터 자동으로 인식한다.
- 상황인지적 모형 선택 : 모형 저장소에 존재하는 모형들 중에서 현재 사용자의 상황에서 사용자가 활용할 가능성이 높은 최적화 모형을 모바일 장비를 통해 우선적으로 보여준다.
- 상황인지적 모형 수립 : 저장되어 있는 최적화 모형 템플릿의 계수나 모수값 중 일부를 현재의 사용자 위치와 같은 상황 정보로부터 받아 최적화 모형을 완성한다.
- 상황인지적 모형 실행 : 모형을 실행하는 문제해결자(solver)가 복수개 존재하는 경우 현재의 무선 네트워크 및 사용자 위치에 따라 어떤 것을 선택하는 것이 더 좋은지를 결정하거나, 사용자의 현 상태에 따라 최적화 알고리즘을 사용할 지 신속하게 해를결정하는 발견적 기법을 사용할 지를 결정하여 실행한다.
- 상황인지적 해의 해석 : 사용자가 사용 중인 모바일 단말기의 성능에 따라 해석 정보의 양을 조정하여 제시한다.

상황은 특정 개인을 둘러싼 모든 직간접적 또는 잠재적인 자극 요소들의 총칭으로서, 크게 어떠한 행동에 영향을 줄 수 있는 물리적이거나 사회적 환경을 의미하는 상황과 외부적으로 들어나지 않는 정신적 상태로 사용자의 행동을 기초로 한 추론을 통해 알아 낼 수 있는 상황으로 나누어진다[7, 11]. 또한 <그림 1>과 같이 시간(Time) 상황 외에 대상(Identity), 위치(Location), 실체(Entities) 상황 등으로 분류하고 이 네가지 종류의 상황을 모아 TILE이라 지칭하기도 한다[1]. 첫째, 시간에 대한 상황이 사용되는 경우는 사용자가 어떠한 서비스를 제공되는 지역에 들어 갔을 경우 현재 그 서비스가 제공되는 시간이면 서비스가 제공되고 그렇지 않을 경우 반응하지 않는 것을 들 수 있다. 둘째, 대상이라고 하는 상황은 현재 사용자가 누구와 함께 있는가에 대한 정보다. 셋째로, 위치는 사용자의 현재 위치 및 관심있어 하는 대상의 위

치에 대한 것이다. 마지막으로 실체에 대한 상황은 현재 사용자가 위치한 위치에 사용 가능한 또는 하드웨어가 센싱 가능한 형태의 물체들에 대한 또는 그 물체가 보유한 정보이다. 그리고 이러한 상황들이 복합적으로 활용되어 유도 가능한 복합적인 상황(compound context)도 존재할 수 있다. 예를 들어 사용자의 활동(activity)은 사용자의 위치와 실체에 대한 상황이 복합되어 유도될 수 있다.

상황은 그것이 사용자의 입력에 의하지 않고 다양한 센서에 의하여 자동으로 인지될 때 서비스의 질을 향상시킬 수 있으며, 또한 자동화(automated) 또는 선용적(proactive)인 서비스를 가능하게 한다. 이를 가능하게 하는 것이 상황인지적 컴퓨팅(context-aware computing)이며, 본 연구에서 제안하는 방법론에 포함시키고 있다.



<그림 1> 상황인지적 모형화

본 연구에서는 위의 다섯가지 상황인지적 모형화 단계 중에서 최적화 모형에 대한 상황인지적인 선택과 모형의 수립에 초점을 두었다.

### 3.1 최적화 모형의 상황인지적인 선택

본 연구에서 고려하고 있는 사용자의 상황은 Abowd [1]의 분류에 따랐으며 구체적으로 다음과 같이 표현된다.

```
Context ::= Time_context+Identity_context
           +Location_context+Entity_context ;
Entity_context ::=Calendar_context ;
Calendar_context ::=Calendar_type+Calendar_content ;
Calendar_type ::=["class" | "home" | "meeting"
                 | "meal" | "leisure" | "misc"] ;
Time_context ::= Timestamp + (Date + AM/PM
                              +Hour + Minute + Second) ;
Location_context ::= Zipcode + Building_name ;
Building_name ::= String ; /* acquired from location
                           ontology */
```

다음은 모형 선택 규칙을 생성할 수 있는 테이블에 대한 메타데이터이다.

```
Model_Selection ::= Record_ID+Condition+Conclusion ;
Condition ::= Time_Context+Location_Context
             +Calendar_Context ;
Conclusion ::= Model_ID+Model_Description ;
```

<그림 2>는 위의 메타데이터에 의해 작성된 모형 선택 규칙 테이블이다. 이 테이블의 각 레코드들은 사용자 입력 혹은 학습에 의하여 획득 가능하다.

ID	hour	ampm	location	calendar_type	model	model_description
1 7		pm	CEI	class	getWeather	get weather data
3 5		am		meal	getRestaurant	get restaurant recommendation
4 8		pm			getWeather	get weather data
7 5		pm		home	getCurrentLocation	get location data
8 6		pm			getShortestPath	get shortest path
9 7		pm	CEI	class	getShortestPath	get shortest path
10 8		pm			getShortestPath	get shortest path
11 7		pm	CEI	class	getRestaurant	get restaurant recommendation
12 8		pm			getRestaurant	get restaurant recommendation

<그림 2> 모형 선택 규칙 테이블 예

학습 방법에 대해서는 본 연구의 범위를 벗어나므로 생략하기로 한다.

이와 같은 시나리오를 구현하기 위한 상황 모형은 다음과 같은 형태를 갖는다.

```
IF <context(1)>
AND <context(2)>
...
AND <context(n)>
THEN <model(i)>
```

이에 대한 한 예를 들어보자.

(예 1) “Joe는 현재 Wean Hall(WeH)에서 수업을 마치고친구와 점심을 하기 위해 카페로 가려고 한다. 이때 Joe의 스케줄정보(calendar)로부터 그 시간이 수업이 끝나고 식사를 하기 위한 이동시간임을 알고, 수업 장소와 날씨, 그리고 카페의 위치를 고려하여 모형베이스로부터 이를 위해 사전에 준비된 최단경로 네트워크모형을 선택한다. 그리고 그 결과를 Joe의 모바일 장치인 PDA를 통해 알려준다”.

이러한 형태에 기초한 예 1에 대한 상황 모형의 예는 다음과 같다. 이탤릭체는 변수이며 웹서비스나 센서로부터 값을 입력받는다. 이탤릭체가 아닌 것은 사용자가 직접 관장하는 PDA 내 정보이거나 사용자가 수동적으로 직접 입력을 한 경우를 의미한다.

```
IF calendar.class = "finished"
AND calendar.meal = "next"
AND weather.temperature = temperature
AND weather.climate = climate
AND location.current = "WeH"
THEN select.model
= shortest_path_network_model(WeH, restaurant_location,
map, temperature, climate)
```

### 3.2 최적화 모형의 상황인지적 수립

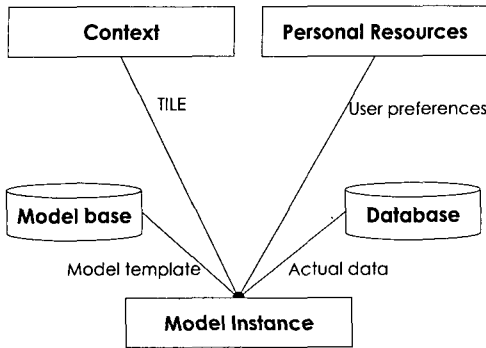
최적화 모형의 상황인지적 수립은 모바일 서비스가 필요한 시점에서 최적화 모형의 인스턴스를

생성하는 과정을 의미한다. 모형 인스턴스란 모형 템플릿에 변수나 모수화 되어 있는 계수 혹은 상수에 값을 대입하여 모형을 실행할 준비가 완료된 상태이다. 따라서 보통의 경우 모형베이스에 들어 있는 모형템플릿에 외부 사용자의 직접 입력 혹은 데이터베이스와 같은 저장소에 저장되어 있는 자료를 추출하여 대입하는 두가지의 방법을 통하여 모형 수립을 수행한다.

그런데 상황인지와 사용자 선호도와 같은 정보가 입수 가능한 경우 다음 <그림 3>과 같이 상황인지적인 최적화 모형 인스턴스가 생성될 수 있다. 모형 템플릿은 마찬가지로 모형베이스에서 선택되어 오되, 3.1에서 언급한 바 있는 최적화 모형의 상황인지적 선택 방법에 의하여 선택되어진다. 그런데 모수 혹은 계수 값은 데이터베이스의 데이터로 결정될 수도 있고, 아니면 인지된 상황 자료에 의하여 결정될 수도 있으며, 또는 이들이 사용자 선호도자료와 결합하여 인지된 자료(perceived data)로 가공된 후에 모수와 계수 값으로 결정될 수도 있다.

예를 들어, 사용자가 현 위치에서 목표지점까지의 최단 경로를 알려주는 서비스를 이용할 경우, 단순한 최적화 모형은 일반적인 물리적 거리를 측정하여 그 경로를 알려줄 것이다. 하지만, 여기에 다가 기온, 날씨 등의 추가적인 상황을 고려한다면 최단 경로의 결과는 바뀔 수 있다. 일반적으로 날씨가 맑을 경우에는 물리적으로 가장 짧은 경로를 통해서 걸어가면 되지만, 비가 올 때마다 기존의 최단 경로 사이에 물이 차서 그 곳을 통과하지 못하게 되는 경우에는 그 곳을 피해갈 수 있는 최단 거리를 구해야 한다. 또는 날씨가 더운 날, 그들이 있는 거리와 그들이 없는 거리가 있을 수 있는데, 그들이 없는 거리는 실제로 사용자가 걸어갈 때, 그 거리가 더 길게 느껴지며 걸음걸이가 힘들 수 있으며, 반면에 그들이 있는 거리는 실제 물리적 거리는 길더라도 시원하게 걸어갈 수 있기에 사용자가 걸어갈 때 상대적으로 짧은 거리로 인지되고 적은 힘을 들여 걸을 수 있다. 즉, 최적화 모형에 상황을

추가적으로 고려할 때, 인지된 값(perceived value)을 계산해 낼 수 있다.



〈그림 3〉 모형의 상황인지적 수립

구체적으로 다음과 같이 한 예를 들어보자.

(예2) “Joe는 통상 연구실을 벗어나 어떤 카페에 가서 점심 식사를 하기 위해 위의 예 1에서 선택된 최단 경로 모형을 가지고 최적 경로를 찾기로 했다. 그런데 밖에 눈이 오고 바람이 심하게 불고 있다. Joe의 현재 위치와 카페의 위치, 그리고 날씨(기후, 온도)에 대한 정보를 얻어서 카페를 가는 최단 경로를 선택하려고 한다. 이때 만족도 극대화를 위한 목적함수 식에는 날씨가 고려된 인지된 거리가 목적함수 계수로 존재한다.”

위의 모형 예로 아래와 같은 간단한 최단경로 모형의 템플릿을 보자.

$$\begin{aligned}
 & \text{Minimize} \\
 & z = \sum_i \sum_j C_{ij} X_{ij} \\
 & \text{s. t.} \\
 & \sum_j X_{oj} = 1 \\
 & \sum_j X_{jn} = 1 \\
 & \sum_i X_{ij} = \sum_k X_{jk} (\forall j) \\
 & X_{ij} = 0, 1 (\forall i, j)
 \end{aligned}$$

여기서 인덱스 0은 사용자의 현재 위치를, N은 도달하려고 하는 목표 지점의 위치를 의미한다. 또한 각 경로 별 비용  $C_{ij}$ 는 인지된 거리(perceived distance,  $PD_{ij}$ )의 함수로 가정한다. 이때 인지된 거리  $PD_{ij}$ 는 다음과 같이 계산된다.

$$PD_{ij} = \alpha_{TILE} * AD_{ij},$$

단 여기서  $\alpha_{TILE}$  는 특정 상황(T, I, L, E)에서의 사용자의 선호도 정보에 의해서 도출된 조정 계수이며,  $AD_{ij}$ 는 유클리디언 방법으로 얻어지는 실제 물리적 거리(actual distance)이다.

요약하면 위의 모형템플릿이 인스턴스화 하기 위한 정보들은 각각 <표 1>과 같은 원천으로부터 나오게 되는 것이다.

〈표 1〉 정보 내용과 원천

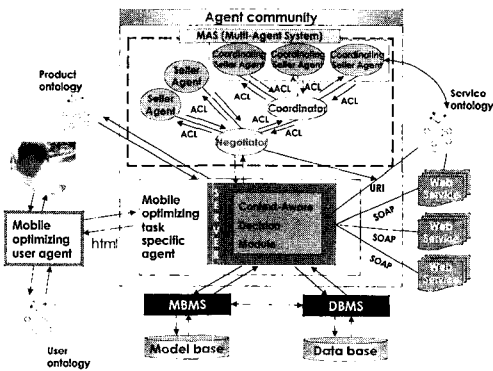
정보 내용	정보 원천
T, I, L, E	상황
$\alpha_{TILE}$	개인 정보원천 (personal resources)
$AD_{ij}$	데이터베이스
$  \begin{aligned}  & \text{Minimize} \\  & z = \sum_i \sum_j C_{ij} X_{ij} \\  & \text{s. t.} \\  & \sum_j X_{oj} = 1 \\  & \sum_j X_{jn} = 1 \\  & \sum_i X_{ij} = \sum_k X_{jk} (\forall j) \\  & X_{ij} = 0, 1 (\forall i, j)  \end{aligned}  $	모형베이스

결국 본 연구에서 사용된 서비스의 최적화 모형은 모든 상황이 고정되어 결과를 추천하는 단순한 최적화 모형에서 벗어나, 현재의 최적화 모형에 또 새롭게 상황을 적용하여 최적화 결과를 계산해 내는 모형이다.

### 3. CAMA-myOptimization

본 연구에서 제안하고자 하는 사용자 상황 인지 기반의 모바일 서비스 최적화 프레임워크는 <그림 4>와 같다. 처음에 사용자는 PDA 등의 모바일 장치를 이용하여 서비스를 제공하는 서버에 접속한다. 사용자가 서비스를 이용할 때 TSA는 모형 베이스에서 해당 서비스에 맞는 최적화 모형을 불러오게 된다. 예를 들어, 사용자가 이용하는 서비스가 거리의 이동에 관련되어 있을 경우에는, 이 서

비스를 작동하기 위해 필요한 변수, 즉 현재의 위치, 목적지, 그리고 가장 짧은 거리 계산에 영향을 줄 수 있는 날씨 등을 읽을 수 있는 최적화 모형을 불러온다. 실제로 기온이 높거나, 비가 오거나, 눈이 오는 상황의 여부에 따라 이동 가능한 거리가 바뀌거나 인지되는 거리가 다르게 느껴질 수 있으므로 이 서비스에서 날씨의 변수도 중요하게 작용한다. 현재의 위치는 GPS나 WiFi 기반의 AP(Access Point), RFID, 또는 블루투스 기반으로 추적하거나 또는 목적지와 마찬가지로 사용자로부터 직접 입력을 받을 수도 있다. 날씨 정보는 이미 구축된 다른 웹서비스를 이용하여 불러올 수 있다. 만약, 사용자가 쇼핑 웹서비스를 이용하게 될 경우에는 또 다른 최적화 모형을 모형베이스에서 불러오게 된다. 즉, 사용자가 쇼핑 서비스를 선택하면 사용자가 원하는 스타일의 의류, 의류의 종류, 희망하는 가격대, 의류의 색깔 등의 변수를 적용할 수 있는 최적화 모형을 불러오게 된다.



〈그림 4〉 상황인지 기반의 모바일 서비스 최적화 프레임워크

PDA에는 해당 PDA 사용자의 UA가 접근할 수 있다. 이 에이전트는 사용자 온톨로지(user ontology)에서 사용자에 대한 정보를 읽은 후 사용자의 요구사항을 반영하고자 이 정보를 사용자의 이동에 맞게 상황인지적으로 최적화를 가능케 하는 업무수행 에이전트(Mobile optimizing task specific agent : 이하 TSA)에게 전달한다. UA가 사용자의

요구사항을 의사결정엔진에 전달했을 때, TSA는 사용자의 요구사항에 맞는 최적화 모형을 찾게 된다. 최적화 모형을 찾기 위해서 모형베이스관리시스템(MBMS : Model Base Management System), 그리고 데이터베이스관리시스템(DBMS : Data Base Management System)와 연결된다. TSA는 해당 서비스에 필요한 최적화 모형을 모형베이스로부터 불러오게 되는데, 이 때에는 모형베이스관리시스템을 거쳐서 불러오게 된다.

TSA는 필요한 경우 다중 에이전트시스템(Multi-agent system : 이하 MAS)와 대화하도록 구성되어 있다. TSA 내부에는 상황 인지적 의사결정 모듈(Contextst-aware decision module : 이하 CDM)이 있다. CDM은 사용자 활동 상황과 그 밖의 기온, 날씨 등의 상황을 추가로 고려한 의사결정 알고리즘이다. 서버는 항상 제품 온톨로지(product ontology)에 있는 정보와 제품을 판매하는 업체들의 에이전트와 연결하여 서비스를 제공한다. TSA를 통해 사용자 정보를 받게 되면, 의사결정엔진에서는 이 정보를 MAS에 있는 협상자에게 보낸다. 그러면 협상자는 이 정보를 다시 판매자 에이전트(Seller Agent : 이하 SA)에게 보내거나 조정자(Coordinator)에게 보낸다. 또 다시 조정자는 자신이 조정을 중재하는 SA들에게 이 정보를 보낸다. 이 정보에 대해 판매 의사가 있는 SA들은 협상자 또는 조정자에게 자신의 판매의사를 보이게 된다. 그런데, 여기에서 조정자는 여러 SA들 간의 조정을 중재하여 그 해당 사용자 정보에 대해 가장 유리하게 제안한 SA의 제안 내용을 협상자에게 추천하게 된다. 최종적으로 협상자에게 수집된 정보는 사용자 온톨로지에 있는 선호도 정보 등을 토대로 하여 최상위 몇 개만이 추천되어 서버의 의사결정엔진으로 보내어진다. 추천되는 개수는 기본적으로 정해지기도 하고, 사용자가 임의대로 정할 수도 있다. 이 과정에서 협상자에 의해 추천되는 정보는 날씨, 기온, 거리 등의 웹서비스를 참고하여 각 상황에 따라 사용자의 효율을 높여줄 수 있는 목록이 바뀌게 된다. 이때에 CDM은 협상자

의 추천정보와 상응되는 웹주소를 이용하여 그 웹 서비스들의 서비스 온톨로지들을 불러온다. 그리고 CDM은 현재 사용자에게 추천될 정보를 HTML의 형식으로 UA에게 전달하게 되고 이것이 사용자에게 보여지게 된다.

### 3.1 시맨틱 웹 서비스

본 연구의 모바일 서비스의 핵심적인 특징은 단순히 정보만 수동적으로 제공하는 것이 아니라 사용자의 상황을 인지, 그에 맞는 최적의 정보 또는 최적해를 제공하는 것이다. 이러한 취지에서 볼 때에 사용자와 서비스에 대한 정보가 공유되고, 위치, 날씨, 일정과 같은 사용자의 상황 정보를 활용하여 모바일 장치를 통해 고부가가치의 개인화된 모바일 서비스를 시맨틱 웹상에서 제공하는 것은 매우 유용할 것이다. 그래서 본 연구에서는 시맨틱 웹서비스를 채택하였다. 시맨틱 웹서비스는 지능형 서비스 자동화를 위해 복수개의 웹서비스들이 서로 교환하는 개념에 대한 지식을 공유, 이해함으로써 처리의 정확도를 기하기 위한 웹서비스 기술로서 주로 프레임 기반의 온톨로지를 활용하고 있다.

한 가지 예를 들어보자. 무선랜에 기반을 둔 PDA가 시맨틱 웹에 접근가능 하도록 설계하면, 사용자의 위치 정보(무선랜을 기반으로 한 위치추적 기술을 이용), 스케줄 정보, 기상 정보, 상황 기반 선호도(예를 들어, “일을 하고 있을 때에는, 홍보용 메시지로 인한 방해를 받고 싶지 않다”, “비가 내릴 때에는, 나는 밖에 나가고 싶지 않다”, “날씨가 맑고 4시가 지났다면, 나는 내가 있는 회사에서 창밖으로 길거리 행사를 볼 수 있다.”)를 포함한 개인 선호도와 사회적 상황 정보가 쉽게 결합될 수 있을 것이다. 그리고 이러한 결합은 곧, 사용자 상황인지가 가능한 최적의 모바일 서비스를 제공할 수 있을 것이다. 따라서, 본 연구에서 제안하는 방법론은 향후 시맨틱 웹이 지향하여야 할 새로운 분야를 제안하는 것이라 할 수 있다.

한편, 웹서비스는 사용자에게 동적인 서비스 제

공을 하기 위한 필수 요소 중의 하나이다. 그러한 서비스를 위해서 <그림 4>에서 보는 바와 같이 서버는 XML에서 오브젝트들간에 메시지를 원활하게 교환할 수 있도록 지원하는 프로토콜인 SOAP(Service Oriented Access Protocol 또는 Simple Object Access Protocol)을 사용하여 여러 웹서비스들과 연결되어 있다.

본 연구에서 활용하고 있는 웹서비스는 다음과 같다.

첫째, 각 지역별, 그리고 날짜 별로 날씨 및 기온 등에 대한 정보를 제공해 주는 웹서비스(weather web service),

둘째, 각 건물 또는 지역의 위치에 대한 정보를 제공해 주고 또한 두 지점에 대한 정보를 받아 그 지점 사이의 거리를 계산하는 거리 계산 웹서비스(distance calculation web service),

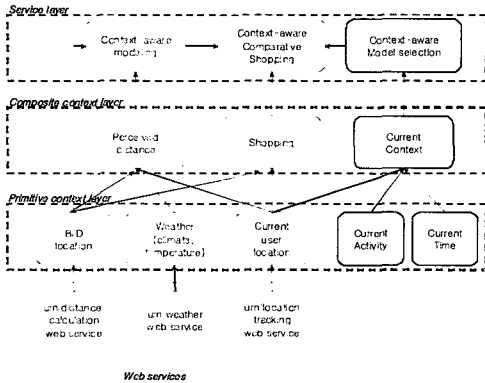
셋째, 사용자의 위치를 추적하여 얻어진 위치 정보를 온톨로지의 형태로 저장하고 그 정보를 공유하여 주는 위치 추적 웹서비스(location tracking web service),

넷째, 심플렉스 알고리즘을 탑재하여 매트릭스 형태의 모형 인스턴스를 받아 최적해를 제공하는 선형 문제 해결자 웹서비스(LP solver web service).

다섯번째, 사용자의 비교 구매를 위한 조정 활동에 참여하는 판매자 웹서비스들(supplier web services)이 중요한 웹서비스로 사용되고 있다.

위의 다섯가지 웹서비스 중에서 앞의 세가지 웹서비스는 주로 정보를 제공하는 정보 제공 지향형 웹서비스(information oriented web service)이며, 뒤의 두가지 서비스는 위탁받은 문제를 해결하는 문제 해결 지향형 웹서비스(problem solving oriented web service)이다. 이 중에서 정보 제공 지향성 웹서비스들은 기초적(primitive)인 상황 혹은 정보를 제공하는 것으로부터 시작하여 복합적(composite) 상황을 유추하기도 하고, 그 결과를 문제해결 지향형 웹서비스나 여타 서비스에 제공하는 일을 하는데, 그 구조의 예는 다음 <그림 5>와 같다.





<그림 5> 정보제공 지향형 웹서비스의 서비스 구조

또한 판매자 웹서비스는 서비스 온톨로지에 해당 정보를 소유하고 있어서 서버에서는 해당 웹소스를 통해 언제든지 공유가 가능하다. 이러한 웹서비스는 사용자가 구매 서비스를 이용할 때, 해당 상점의 위치, 상점에 도달할 수 있는 경로를 날씨 및 기온, 그리고 사용자의 사회적 상황을 포함한 다양한 상황에 따라서 매 순간마다 사용자에게 가장 적합한 정보의 제공을 가능하게 해 준다. 부록 A는 한 서비스 온톨로지의 예이다.

### 3.2 다중 에이전트 설계 및 작동

본 연구에서 제안하는 사용자 상황인지 모바일 서비스는 다중 에이전트간의 상호협동과 협상의 방법으로 수행되도록 설계되어 있다. 이러한 다중 에이전트 설계는 에이전트간의 원활한 메시지 교환을 위하여 프로토콜이 필요하고(본 연구의 경우 SOAP), 아울러 시맨틱 웹서비스 차원에서 제공될 수 있도록 설계되었다. 본 프레임워크에 포함되어 있는 UA, TSA, 그리고 CSA 중에서 UA와 CSA를 상세히 설명하고자 한다.

#### 3.2.1 사용자 에이전트(UA)

UA는 자신의 사용자를 위해 비교구매를 위한 요청을 협상자(negotiator)에게 할 수 있고, 협상자로부터 온 결과를 선택할 수 있다. UA의 협상자와

의 의사소통 알고리즘은 다음 <그림 6>과 같이 개발되었다.

Algorithm for consumer  $i$ :

Initiate initial offer  $D_0$  and send it to the negotiator

Repeat

Receive  $D$  from the negotiator

Announce to the negotiator a consumption plan  $D \in R_+^n$  that

maximizes  $u_i(x_i) = u_i(p, q, c, d)$  given the budget constraint, where  $p, q, c$  and  $d$  denotes price, quality, condition, and distance, respectively

Until informed that an equilibrium has been reached

Exchange and consume

<그림 6> UA의 의사소통 알고리즘

#### 3.2.2 조정판매자 에이전트(CSA)

조정은 모바일 커머스 문제에 요구되는 요소이다. 한 시스템 내에서 여러 에이전트들이 활동을 하게 될 때에는 그 에이전트들 간에 서로 의사소통을 하며 조정할 수 있는 메커니즘이 필요하게 된다. 그러므로 다중 에이전트와 조정(Coordination) 간의 관계에 대한 분석이 필요하게 된다. “조정(coordination)”이라는 용어에 대해서는 다양한 정의가 발견된다[4, 5, 6, 8, 9, 12, 13, 16, 17, 19]. 본 연구에서는, 조정은 의존성을 관리하는 것의 활동과, 협동적인 활동에 상호 관련된 공통 업무들에 연루된 협동 개체들 간에 발생할 수 있는 충돌들을 해결하는 활동을 위한 지원시스템이라 정의한다. 이러한 정의가 모바일 커머스 문제에 적용되었을 때, 개체들은 각각 구매자와 판매자를 나타내는 에이전트들을 참조하고, 이 에이전트들은 구매자와 판매자 양쪽 모두에게 가장 좋은 거래를 성사시키고자 함께 작동한다. 다중에이전트 간의 협동적인 행동은 다중에이전트 조정을 지원하기 위해 하나의 중앙 조정 에이전트(central coordination agent)로 통합함으로써 더욱 진보될 수 있다[2, 3, 13, 15, 18].

중앙 조정 에이전트로서 본 연구에서 제안하는 조정판매자 에이전트인 CSA는 웹서비스들을 통해

제안된 여러 제품들을 비교하기 원하는 사용자에 이진트인 UA에 항상 반응을 한다. UA는 사용자의 모바일 장치에 다운로드 받아서 사용할 수 있다. 사용자가 쇼핑을 가거나 구매하고자 하는 제품을 찾을 때, 그 사용자는 경쟁업체에 있는 다른 유사한 경쟁력 있는 제품들을 찾는 것을 도와줄 자신만의 전용 UA를 얻게 될 것이다. 그런 후에 UA는 조정자에게 하나의 새로운 요구사항을 전달하여 조정자로 하여금 UA가 제시한 것과 동일한 혹은 더 나은 조건의 유사한 제품을 제시할 의향이 있는 다른 SA들을 소개하도록 한다. 조정자는 먼저 스스로 만족된 정보 저장소를 질의함으로써 관련 있는 CSA들의 집합을 선택한다. 그런 후에 UA로부터 전달 받은 요구사항은 선택된 CSA들에게 전달된다.

이 시스템에서의 협상의 최종 목표는 UA와 CSA 사이에 존재하는 윈-윈의 상태를 인지하는 것이다. UA는 사용자가 모바일 서비스를 이용하는 그 시간에 실제로 보게 되는 것보다 경쟁력 있는 제품들을 더 많이 알게 될 것이다. CSA는 조정자에 의해 창출된 새로운 사용자, 즉 구매자들의 요구를 완수함으로써 총 매출과 이익을 증대시킬 수 있다. 또한 조정자는 협상자를 거쳐서 UA와 CSA 간의 거래 성공률을 극대화할 것이다. 이것을 달성하기 위해서 조정자는 다음과 같은 역할을 수행한다.

- 목표 1) 사용자가 원하는 것에 대한 정보를 가능한 자세하고 정확하게 제공해야 한다.
- 목표 2) CSA가 좀더 사용자의구매의욕을 높일 수 있는 매력적인 제안을 할 수 있도록 한다.
- 목표 3) CSA로부터 제안되는 내용 중, 사용자 선호도, 총 이익, 그리고 협상에 성공할 확률에 의하여 가장 좋은 제안을 선택한다.
- 목표 4) 협상자를 통하여 UA에게 가장 좋은 제안을 전달한다.

이중에서 특히 목표 1)은 CSA가 사용자의 선호에 얼마나 정확히 적합한가와 밀접하게 관련이 있다. 그러나, CSA는 수많은 사용자들의 프로파일들을 미리 아는 것은 거의 불가능하기 때문에 정확

한 사용자들의 효용 함수는 모른다. 이 문제는 이전의 판매정보를 이용함으로써, 즉 조정자에 Case base를 이용함으로써 해결할 수 있다. 그리고 목표 2)를 달성하기 위해서는, 요구사항을 받아들이는 각각의 CSA는 자신의 판매 순이익을 최적화하기 위해서 자신의 비용함수에 관련된 의사결정 매개변수를 바꾸면서 비용/편익분석을 하도록 해야 한다. 이에 관련한 최적화 모형은 다음과 같을 것이다.

Maximize Total\_Profit = Sales Volume \*(Unit Price - Unit Cost)  
 Subject to :  
 Suggesting condition must be within the delegation area.  
 Suggesting condition must be better than any other conditions made by any other CSA and initial condition

협상을 자동화하기 위해서는 각각의 CSA는 자신의 사용자에 대해 융통성을 갖추고 있어야 한다. 예를 들어, 어떤 사용자가 허용할 수 있는 가장 낮은 가격이 알려질 수 있고 이에 반응하는 CSA는 자신이 허용할 수 있는 가격 레벨에 영향을 주지 않는 선에서 자신의 가격을 바꾸어주면서 협상을 할 수 있다. 다음 <그림 7>은 협상 알고리즘이다.

Algorithm for CSA k :  
 Repeat  
     Receive current advertisement( $D_L^*$ ) from the Coordinator  
     Apply fast search algorithm  
     Announce to the Coordinator an advertisement ( $D_k$ ) that maximizes  
          $\pi_k = p_k = UC_k$   
     Until informed that an equilibrium has been reached  
     Exchange and produce  
 Fast search algorithm for CSA k :  
     Estimate preference of best advertisement  
     If the preference is better than the preference of  $D_L^*$ , proceed to next  
     Otherwise, abstain offering  
 Repeat  
     Set quality and condition as best  
     Acquire estimated preference by changing price only  
     Until estimated preference becomes higher than the preference of  $D_L^*$

<그림 7> CSA의 협상 알고리즘

## 4. 구 현

본 연구에서는 3장에서 언급한 프레임워크에 기초하여 CAMA-myOpt라고 명명된 프로토타입 시스템을 구현하였다. CAMA-myOpt은 사용자 상황을 인지하여 주어진 조건하에서 최적화하여 사용자의 효용을 극대화 할 수 있는 모바일 서비스 제공을 위한 플랫폼이다. CAMA-myOpt에서 사용된 사용자 온톨로지, 제품 온톨로지, 서비스 온톨로지 등은 XML을 기반으로한 DARPA 에이전트 마크업 언어인 DAML(DARPA Agent Markup Language)+OIL로 구현되었다. 그리고, UA, TSA, SA, CSA 등은 JDK 1.4.1로 구현되었으며, 최종 사용자 인터페이스는 JSP로 구현하였다. 협상자에서 SA와 조정자로의 의사소통과 조정자에서 CSA로의 의사소통은 FIPA-ACL로 구현되었다. 그리고 TSA의 활동 환경에 있는 서버와 에이전트 커뮤니티 외부에 있는 웹서비스 간의 의사소통은 Apache SOAP 2.0 Server로 구현되었다. 또한 CAMA-myOpt에서 사용하는 모험베이스, 데이터베이스 등을 관리하기 위해서 Jet Engine을 이용하였다.

### 4.1 상황인지

사용자는 PDA를 이용하여 서버에 접속하게 된다. 서버에 접속하게 되면, 사용자는 자신에 대한 정보를 저장할 수 있는 페이지와 자신이 지금 이용하고자 하는 서비스 페이지에 접근할 수 있다. 사용자가 상황질의 페이지로 이동하게 되면 사용자 자신의 프로파일용 저장할 수 있는 메뉴, 사용자 자신의 선호도를 저장할 수 있는 메뉴, 그리고 현재의 상황을 변경할 수 있는 메뉴가 있는 페이지를 접하게 된다. 이 페이지는 사용자의 PDA의 스크린에 <그림 8>의 (a)와 같은 형태로 보여지게 된다.

UA는 사용자가 자신의 PDA에 다운로드 받은 후 설치할 수 있는 하나의 소프트웨어 프로그램이며, UA는 이 페이지에서 사용자가 지정하는 사용자의 프로파일, 선호도, 그리고 상황에 관한 정보

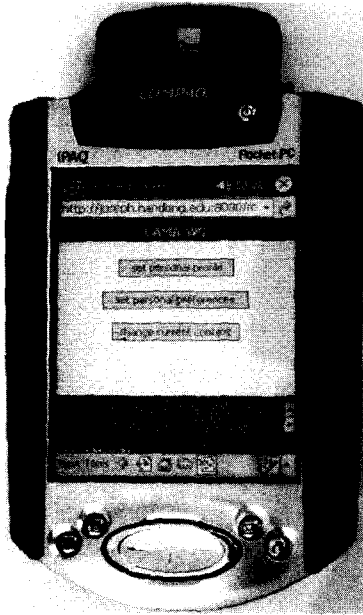
를 읽은 후, 사용자가 원하는 서비스에 접속을 할 때에 그 정보들을 토대로 TSA에게 전달하여 사용자에게 가장 적합한 정보를 추천 받을 수 있도록 도와주는 역할을 하게 된다.

### 4.2 상황인지적 모형 선택

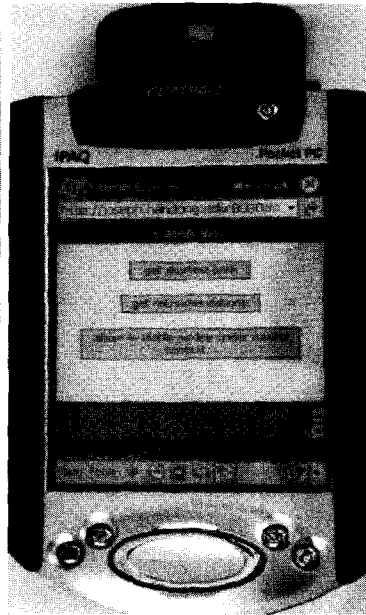
사용자는 서버에 접속하였을 때, 그 서버에서 지원 가능한 모바일서비스 중에서 자신이 원하는 서비스를 선택할 수 있다. 그런데, 사용자가 선택하는 서비스의 종류에 따라서 사용자에게 추천할 정보를 최적화하는 모형은 다르다. 즉, 사용자의 상황에 따라서 그 해당 모바일 서비스에 적용할 수 있는 최적화 모형이 바뀌게 된다. 예를 들어, 사용자는 <그림 8(b)>에서와 같이 현재의 위치에서 목표 지점까지 가는 경로 중 가장 짧은 거리를 구하는 서비스(get shortest path), 또는 백화점에서 의류를 구매하고자 다니면서 어떠한 브랜드의 어떠한 것이 가장 좋은지 추천해 주는 서비스(get recommendation)를 선택할 수 있다. 그리고 각각의 상황에 따라서 어떠한 최적화 모형이 가능한지 보여주는 메뉴(show available models under current context)도 있다.

### 4.3 상황인지적 모형 수립

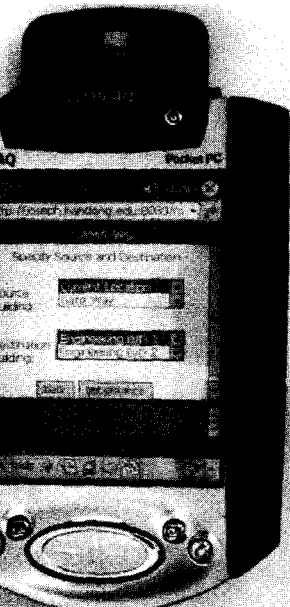
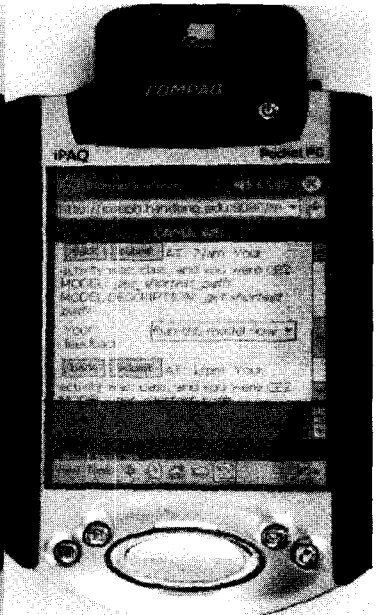
사용자가 최단 경로를 구하는 모바일 서비스를 선택하게 되면, 서버에서는 모형베이스관리시스템을 통해 기존에 만들어진 최적화 모형 중에 이 서비스에 가장 적합한 최적화 모형을 선택하여 사용자에게 보여준다. 그러면 사용자는 자신이 원하는 결과를 얻기 위해 몇 가지 값을 입력하게 되는데, 최단 경로를 구하는 이 서비스의 경우에는 자신의 현재 위치와 이동하고자 하는 목표지점을 입력하면 된다. 실외의 경우 GPS, 실내의 경우 AP(Access Point)를 이용하여 사용자의 위치가 인지되는 경우에는 현재의 위치에 대한 입력 박스가 없고 목표 지점만 입력하면 되는 최적화 모형이 선택된다(<그림 8(c)> 참조).



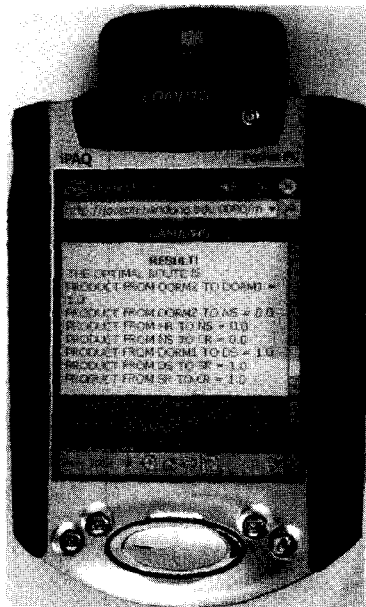
(a) 상황 질의



(b) 상황인지 모형 선택과 실행



(c) 상황인지 모형화



(d) 상황인지 모형 인스턴스 생성

<그림 8> 구현 PDA 스크린

사용자가 현재 위치와 목표지점에 대한 정보를 서버에 제출하게 되면 서버에서는 사용자의 현재 위치와 목표지점을 인지하여 위치정보 웹서비스와 날씨정보가 웹서비스 등의 서비스들을 방문하여

해당 정보를 불러오게 된다. 이렇게 얻어진 사용자 정보와 상황 정보를 종합적으로 고려한 후에 사용자에게 최단 경로에 대한 정보를 보여주게 된다. 물론, 사용자에게 추천되는 최단 경로는 물리적 거리

가 아닌 인지된 거리를 고려하여 계산된 결과이다.

본 연구에서는 학교 교내의 건물들에 대한 위치를 x좌표, y좌표로 표현하여 위치를 계산하고 그 위치에 대한 정보를 데이터베이스에 저장하였다. 이 데이터베이스의 위치정보는 거리를 계산하는 웹서비스가 각 건물마다의 거리를 계산해 내는 데 유용하게 쓰인다. 그리고 사용자의 현재 위치에서부터 목표지점까지의 경로 중에서 모든 가능한 경로를 CDM이 찾아내어 날씨 웹서비스로부터 현재 위치에 대한 온도와 기후를 얻어온 후에 인지된 거리 개념을 고려하여 계산한 후 최단경로문제를 찾는 최적화 모형을 수립한다. 수립된 모형은 각각 목적함수, 우변상수, 제약식을 표현하는 세가지의 행렬 또는 벡터 형태로 변환되며, 행렬 값은 심플렉스 알고리즘이 내장된 LP문제해결자로 전달되어 최적해를 구하게 한 다음, 그 최적해를 다시 벡터값으로 받게 되면 CDM은 다시 이를 UA에게로 전달한다. 결국 PDA의 사용자는 인지된 최단 경로를 기반으로 하여 걸어가야 할 건물 사이의 경로를 보게 되는데, 이때 가게 될 건물은 1.0, 피해야 할 건물 사이의 경로는 0.0으로 표시된다. 이에 대한 PDA에서의 디스플레이는 <그림 8(d)>에 나타나 있다.

#### 4.4 시스템 성능

구현상의 제약으로 현재 구축된 CAMA-myOpt는 실시간 환경에서의 실행에 있어서 다음과 같은 기술적인 문제를 가지고 있다. 첫째는 서비스 반응 시간의 문제이다. 상황인지적 모형화 인식의 단계에서부터 해의 해석을 사용자 인터페이스로 보이는 과정에서 다음과 같은 시간이 소요된다.

- 전체적인 소요 시간 = 최적화 서비스 요청이 사용자 장비에서 서버로 전송되는 시간 + 사용자 상황을 인지하기 위해 관련 웹서비스로부터 정보를 받는데 소요되는 시간 + 사용자 정보로부터 모형 선택 규칙을 실행하는 시간 + 선택된 모형의 템플릿과 관련 인스턴스 자

료를 최적화 웹서비스에서 수행하는 시간 + 수행된 최적해를 사용자 장비로 전송하는 시간

이중에서 하나 이상의 DAML+OIL 형태의 온톨로지 인스턴스 파일들에 접근하여 다운로드하고 이를 파싱하여 활용하는데 있어서 다소 시간이 지체되는 경향이 있으며, 결국 반응속도에 영향을 주는 것이 발견되었다. 그러나 이는 서비스 반응시간과 정보의 공유로 인한 서비스 질의 향상과의 상충성이 있으므로 시스템의 문제라기 보다는 서비스설계의 이슈일 것이다.

또한 최적화 모형을 생성하여 실행하는 과정에서 변수나 모형이 커지는 경우 최적화 도출시간이 증가하는 경우 속도는 더 느려질 수 있다. 그러나 본 연구는 최적화 알고리즘의 성능을 개선하는 것에 주안점을 주지 않았으므로, 기존의 알고리즘을 최적화 웹서비스에서 사용한다고 할 때 다음과 같은 몇가지 방법으로 완화할 수 있을 것이다. 첫째는 서비스지역(service zone)을 캠퍼스 지역 등과 같이 제한하는 것이다. 현재 준비 중인 대부분의 위치 기반 모바일 서비스나 유비쿼터스 서비스가 주로 실내(in-door) 서비스나 제한된 서비스 지역을 상정하는 것을 볼 때 본 연구의 방법을 적용하는데는 큰 문제가 없으리라고 본다. 실제로 본 연구의 실험도 캠퍼스를 대상으로 했기 때문에 이 부분에서의 속도 문제는 없었다. 둘째는 변수의 입도(granularity)를 상황인지적으로 조절하는 방법이다. NavTech에서 제공하는 야후맵과 같이 범위가 광역화되는 경우 인식하는 노드의 수를 줄이는 것과 같은 이치이다. 셋째는, 서비스 지역을 의미 있는 몇가지의 구역으로 분해한 후에 지역 최적화를 실시하고 후에 각 최적해를 연결하는 방법이다. 이때 각 최적화는 서로 다른 웹서비스에서 실시하므로 다중 프로그래밍의 효과가 있어 시간을 줄일 수 있다.

## 5. 결론 및 추후 연구방향

본 연구에서는 유비쿼터스 환경이 현실화 되어

가고 있는 현재 상황에서, 상황인지적인 최적화를 지원하는 동적인 모바일 웹서비스를 제공 프레임워크를 제안하고 이를 실제 환경에서 구현하므로써 그 가능성을 검토하였다. 이를 위하여 사용자의 상황인지적인 모형화의 개념을 정리하였고, 이를 위해 상황을 이해할 수 있는 메커니즘의 필요성을 언급하였다. 이러한 메커니즘에는 사용자와 관련된 자료 및 위치, 거리, 날씨 등 다양한 상황을 인지하는 서비스 원천에 접근하는 것을 요구하기 때문에 이를 위해 온톨로지나 혹은 웹서비스로 활용하였다. 이러한 전체적인 체계로 본 연구에서는 다중 에이전트기법을 적용하였다. 이에 UA, TSA, SA, CSA와 같은 에이전트를 설계하고 이를 구현하였다. 특히 상황인지 최적화 모형화를 가능케 하기 위해, TSA내에 있는 CDM에 적용하였다. 이 같은 결과를 토대로 본 연구에서는 DAML+OIL과 자바가상머신(Java Virtual Machine, JVM)을 적용하여 다중 에이전트에 기반한 개인화된 상황인지 모바일 서비스 제공을 위한 플랫폼 프로토타입인 CAMA-myOpt를 구축하여 그 가능성을 검증하였다.

본 연구는 기존의 최적화와 전자상거래와의 상호적 결합 가능성에 대한 논의를 상황인지 능력과 결부하여 모바일 서비스분야로 확장하였다는 데 또다른 학술적 공헌이 있을 것이다. 모바일 서비스의 핵심은 사용자의 상황에 맞춘 서비스를 제공하는 데에 있다. 서비스가 일정한 형태로 고정되어 있거나, 사용자의 정적인(static) 사용자 정보만을 인지하여 제공하는 서비스는 사용자의 요구에 부합하기 어렵기 때문이다. 이에 대하여 본 연구에서 제안한 최적화 모형화를 상황인지적 모바일 서비스에 접목시킨 것은 사용자의 요구사항에 신속히 반응할 수 있게 한다.

또한 상황인지 기능을 활용한 모바일 혹은 모바일과 유비쿼터스를 복합한 모비쿼터스(mobiquitous) 서비스의 상용화를 준비하고 있는 실무자의 관점에서 볼 때, 본 연구는 개인화된 상황인지 모바일 서비스에 어떠한 요소 기술들이 어떻게 연결되어 활용될 수 있는지를 실제적으로 보여준다는 데 의

미가 있다. 특히 최적화 모형을 서비스에 응용하는 것이 과거 의사결정지원시스템(DSS)이나 중역지원시스템(ESS) 외에도 가능하며, 이는 특정 개인을 지원하는 모바일 DSS나 모바일 ESS로 구축의 의미있음도 보이고 있다.

그러나 본 연구는 아직 다음과 같은 몇가지 점에서 4.4절에서 밝힌 성능의 문제 외에 추가적인 연구가 필요하다. 첫번째로, 상황인지적 모형화의 전과정을 모두 지원하는 것이 아니다. 상황인지적 문제 인식과 실행, 그리고 실행된해의 해석에 대해 어떻게 다루어야 하는지에 대해서는 아직 본 논문에서 다루지 않았다. 또한 사용자 수용성의 문제로서 보안의 문제이다. 사용자 자원을 공개하는 것을 DAML+OIL로 웹 상에서 구현하였는데, 아직까지는 정보가 불특정 다수에게 개방되고 있다. 기술적으로 보안은 가능하나, 그러한 경우 공개의 원칙을 가진 시맨틱 웹의 정신에 어느 정도 대치되는 모순을 가지기도 하며, 추후 연구에서 고려되어야 할 것이다.

## 참 고 문 헌

- [1] Abowd, G.D., "Software Engineering Issues for Ubiquitous Computing," *Proceedings of the 21st International Conference on Software engineering*,(1999), pp.75-84.
- [2] Bird, S.D., "Towards a Taxonomy of Multi-Agent Systems," *International Journal of ManMachine Studies*, Vol.39(1993), pp.689-704.
- [3] Bird, S.D. and G.M., Kasper, "Problem Formalization Techniques for Collaborative Systems," *IEEE Transactions on Systems, Man and Cybernetics*, Vol.25, No.2(1995), pp.231-242.
- [4] Bonarini, A. and V. Trianni, "Learning fuzzy classifier systems for multi-agent coordination," *Information Sciences*, Vol.136, No.1-4(2001), pp.215-239.

- [5] Cabri, G., L. Leonardi and F. Zambonelli, "Mobile agent coordination for distributed network management," *Journal of Network & Systems Management*, Vol.9, No.4(2001), pp.435-456.
- [6] Candea, C., H. Huosheng, L. Iocchi, D. Nardi, and M. Piaggio, "Coordination in Multi-Agent RoboCup Teams," *Robotics and Autonomous Systems*, Vol.36, No.2-3(2001), pp.67-86.
- [7] Davies, G. and D. Thomson(1988), "Context in Context," In : Davies, G. & Thomson, D.(eds.) *Memory in Context*, John Wiley, Chichester.
- [8] De Los Angeles Junco Rey, M. and F.R. Quintana, "Improving multi-agent coordination with an approach based on a distributed rational decision-making model," *International Journal of Knowledge-Based Intelligent Engineering Systems*, Vol.5, No.4(2001), pp.268-278.
- [9] Durfee, E.H., "Scaling up agent coordination strategies," *Computer*, Vol.34, No.7(2001), pp.39-46.
- [10] Geoffrion A.M. and R. Krishnan, "E-Business and Management Science : Mutual Impacts (Part 1 of 2)," *Management Science*, Vol.49, No.10(2003), pp.1275-1286.
- [11] Kintsch, W., "The Role of Knowledge in Discourse Comprehension : A Construction-Integration Model," *Psychological Review*, Vol.95, No.2(1998), pp.163-182.
- [12] Kwon, O.B. and K.C. Lee, "MACE : multi-agents coordination engine to resolve conflicts among functional units in an enterprise," *Expert Systems With Applications*, Vol.23, No.1(2002), pp.9-21.
- [13] Kwon, O.B. and N. Sadeh, "Applying case-based reasoning and multi-agent intelligent system to context-aware comparative shopping," *Decision Support Systems*, Vol.37(2004), pp.199-213.
- [14] Murthi, B.P.S. and S. Sarkar, "The role of the management sciences in research on personalization," *Management Science*, Vol.49, No.10(2003), pp.1344-1362.
- [15] Sillince, J.A.A. and M.H. Saeedi, "Computer-mediated communication : Problems and potentials of argumentation support systems," *Decision Support Systems*, Vol.26, No.4(1999), pp.287-306.
- [16] Sugawara, T., K. Murakami and S. Goto, "A Multi-Agent Monitoring and Diagnostic System for TCP/IP-Based Network and its Coordination," *Knowledge-Based Systems*, Vol.14, No.7(2001), pp.367-383.
- [17] Tews, A. and G. Wyeth, "MAPS : a system for multi-agent coordination," *Advanced Robotics*, Vol.14, No.1(2000), pp.37-50.
- [18] Tung, B. and J. Lee, "An Agent-Based Framework for Building Decision Support Systems," *Decision Support Systems*, Vol.25, No.3(1999), pp.225-237.
- [19] Wu, D.J., "Software agents for knowledge management : Coordination in multi-agent supply chains and auctions," *Expert Systems with Applications*, Vol.20, No.1(2001), pp.51-64.

## Appendix A. 서비스 온톨로지의 예

```

<?xml version='1.0'?>
<rdf : RDF
  xmlns : rxsd='http://www.w3.org/2000/10/XMLSchema#'
  xmlns : daml='http://www.daml.org/2001/03/daml+oil#'
  xmlns : rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
  xmlns : restaurant='http://128.2.199.68/restaurants/restaurant-ont#'>
  <daml : Property rdf : about='http://128.2.199.68/restaurants/restaurant-ont#openHourSat' />
  <daml : Property rdf : about='http://128.2.199.68/restaurants/restaurant-ont#buildingLocation' />
  <daml : Property rdf : about='http://128.2.199.68/restaurants/restaurant-ont#restaurantData' />
  .
  --- 중략 ---
  <daml : Property rdf : about='http://128.2.199.68/restaurants/restaurant-ont#maxPrice' />
  <daml : Property rdf : about='http://128.2.199.68/restaurants/restaurant-ont#openHourFri' />
  <daml : Property rdf : about='http://128.2.199.68/restaurants/restaurant-ont#cash' />
  <daml : Property rdf : about='http://128.2.199.68/restaurants/restaurant-ont#foodType' />
  <daml : Property rdf : about='http://128.2.199.68/restaurants/restaurant-ont#openHourSun' />
  <restaurant : restaurant>
    <restaurant : restaurantData>
      <restaurant : restaurantDataPROPERTIES>
        <restaurant : restaurantName
          rdf : type='http://www.w3.org/2000/10/XMLSchema#string'
          rdf : value='Asiana' />
        <restaurant : foodType
          rdf : type='http://www.w3.org/2000/10/XMLSchema#string'
          rdf : value='Chinese' />
        --- 중략 ---
        <restaurant : openHourSun
          rdf : type='http://www.w3.org/2000/10/XMLSchema#string'
          rdf : value='9999' />
        <restaurant : closeHourSun
          rdf : type='http://www.w3.org/2000/10/XMLSchema#string'
          rdf : value='9999' />
      </restaurant : restaurantDataPROPERTIES>
    </restaurant : restaurantData>
    <restaurant : userID
      rdf : type='http://www.w3.org/2000/10/XMLSchema#string'
      rdf : value='RLaurel' />
  </restaurant : restaurant>
</rdf : RDF>

```