

# 소프트웨어 에이전트를 위한 협상 결정함수에 관한 연구<sup>†</sup>

## (A Study on Negotiation Decision Functions for Software Agents)

김 중 한\*  
(Joong-Han, Kim)

**요약** 소프트웨어 에이전트에 의해서, 일상 업무는 자동화 되어왔으며, 그에 따라 사람이 담당하는 업무는 특정 업무범위로 축소되고 있다. 그러나 아직까지 대부분의 에이전트는 전체 업무 프로세스의 일부 단계에서만 그 역할을 수행할 뿐이다. 자동화 업무에서의 주요 단계에서 사용자들을 지원하기 위하여 특정 방향으로 협상 당사자들을 설득하는 데 있어서 소프트웨어 에이전트의 역할은 필수적이다. 사용되는 협상의 종류는 매우 여러 가지이지만, 본 논문은 '협상에 기초한 경쟁적 비즈니스 환경' 과 같은 특정 협상행위에 중점을 두고 있다. 이러한 환경에서의 협상을 위해서 자율적 에이전트는 환경적 변수(예: 협상 경쟁자 수, 각 협상 참여자 수, 협상업무가 완료되는 최대한의 시간, 사용자의 선호도)를 중점적으로 고려하고 있다. 경쟁적 비즈니스 환경에서는 협상진행 중에 끼어들어 계약을 성사시키는 잠재 협상 경쟁자들이 있지만 자동화된 협상을 위하여 제안된 이전의 협상 결정함수들은 단지 시간이나 정해진 수의 협상참여자들만을 변수로 사용해왔다. 따라서 본 논문은 경쟁적 시장환경에서 잠재적 경쟁자에 대한 고려가 포함된 협상 결정 함수와 그러한 고려사항이 포함되지 않은 협상 결정 함수에 대한 기능의 평가를 시도했다. 이 평가를 위해서 본 논문은 다수의 구매자와 판매자가 제한된 자원을 위해 한 공간에서 경쟁하는 전자상거래 시장을 적용 범위로 선택했다.

**핵심주제어:** 소프트웨어 에이전트, 협상, 의사결정함수

**Abstract** Software agents reduce human involvement to a certain extent by automating routine tasks. However, most of agents have assisted with only a few steps in the multi-steps process of electronic transactions. In order to help users with the important steps in the electronic transactions, software agents need to persuade other parties to act in particular ways. While negotiations have many shapes and forms, this paper focuses on a particular class of negotiation, that is competitive business environment based negotiation. For negotiation with other parties in this context, it is necessary for autonomous agents to consider environmental variables - the number of competitors, the number of negotiation parties, the maximum time by which they must finish their jobs, and user's preferences. Previous negotiation decision functions for the automated negotiation have used only time or the static number of negotiating parties as negotiation criteria, although competitive business environment should include potential competitors who can snatch negotiation parties away. This paper attempts to evaluate the performance of a negotiation decision function that considers the potential competitors in competitive market environment as well as that of a negotiation decision function that does not. For this evaluation, this study adopts the electronic marketplace as an application domain because many buyers and sellers compete for limited resources in the marketplace.

**Key Words:** Software Agent, Negotiation, Decision Function

## 1. 서론

경쟁적 비즈니스 환경과 증가하고 있는 업무 및 개인 생활에서의 복잡성은 수많은 직무에 있어서 효율적이고 즉각적인 수행을 요구한다. 이러한 필요성을 충족하기 위해서 새로운 컴퓨팅 도구들이 필요하게 되었다. 지능적 자율성을 가지고 있는 에이전트 개념 역시 이러한 도구 중의 하나이다[1]. 소프트웨어 에이전트에 의해, 일상 업무는 자동화되어가고 있으며, 이에 따라 인간이 관여하는 업무는 특정 업무범위로 축소되어지고 있다. 하지만, 아직까지 대부분의 에이전트는 자동화 업무에서 전체 다단계 프로세스에서 단지 몇몇 단계만을 보조하는 역할만을 할 뿐이다. 소비자의 요구에 따라 제품의 가격을 비교하는 것이 그 예라고 할 수 있다.

전자적 거래의 주요 단계에서 사용자들을 돕기 위해서 한 소프트웨어 에이전트가 업무를 완수할 만한 충분한 능력을 가지고 있지 않고 또한 에이전트들 간에 상호 의존성이 있기 때문에 소프트웨어 에이전트가 다른 에이전트와 상호작용을 하는 것은 필수적이다. 이러한 각 에이전트들은 직접 제어 방식을 가지고 있지 않기 때문에 서로 상대방이 특정한 방향으로 행동하도록 설득해야만 한다. 이러한 종류의 설득행위를 ‘둘 또는 그 이상의 참여자 측에 의해 결합된 결과가 만들어지는 프로세스’, 즉 ‘협상(negotiation)’이라고 부른다[3]. 예를 들어, Kasbah[2]시스템은 거래 프로세스 주요 단계에서 사용자의 편에서 자율적으로 협상하고 거래하는 에이전트를 제공함으로써 사용자들의 편의를 돕는 것이다.

협상은 여러 가지 형태로서 존재하지만 본 논문은 특정 협상 종류, 즉 협상에 기초한 경쟁적 비즈니스 환경에 중점을 두고 있다. 본 논문에서, 자율적인 에이전트는 여러 다른 참여자들과 협상하기 위해 환경적 변수(경쟁자의 수, 협상 참여자의 수, 그들의 업무를 마치기 위한 최대한의 시간, 그리고 사용자의 선호도 등)를 고려한다. 비록 경쟁적 비즈니스 환경에서는 잠재적 경쟁자들이 협상진행 과정에 참여하여 계약을 성사시키기도 하지만 자동화된 협상을 위한 기존의 협상 결정함수들은 단지 시간과 협상 경쟁자의 숫자

만을 변수로 사용했다.

경쟁적 비즈니스 환경은 공급자의 수뿐만이 아니라 수요자의 수까지도 고려해야 한다. 따라서 본 논문은 경쟁적 시장 환경에서 잠재적 경쟁자에 대한 고려가 포함된 협상 결정 함수와 그러한 고려사항이 포함되지 않은 협상 결정에 함수의 역할에 대한 평가를 시도한다. 이 평가를 위해서, 본 논문은 다수의 구매자와 판매자가 제한된 자원을 위해 한 공간에서 경쟁하는 상거래 환경을 적용 범위로 선택했다.

## 2. 소프트웨어 에이전트 기능과 협상결정함수

### 2.1 고객 구매 행위 모델

구매 의사 결정에서 소비자들을 돕는 소프트웨어 쇼핑 에이전트는 여러 가지 형태를 가지고 있다. 표 1에서는, 소비자 구매행위 모델을 6단계로 구분하고 각 단계를 지원하는 대표적인 에이전트 시스템의 예를 보여준다. 본 연구에서는 전자상거래환경에서 소비자들의 구매 프로세스를 이해하기 위해 Guttman의 2인의 연구에서 제시한 소비자 구매행위 모형을 사용했다[9]. 이 구매행위 모형은 수요확인, 제품중개, 상점중개, 협상, 구매 및 배송, 그리고 서비스와 평가의 단계를 포함하고 있다.

수요확인(Need Identification)단계에서는 구매자가 어떤 것이 필요하다고 느끼는 단계로서 예를 들어, 여름이 오면 에어컨을 구매해야겠다고 결정한다. 제품중개(Product Brokering)단계는 여러 제품 중 어떤 특정 제품을 구매할 것인가를 결정하는데 지원하는 기능이다. 이 단계는 소비자가 제공하는 구매기준에 기초하여 대상제품을 평가하고 결과물로서 특정제품을 산출한다. 상인 중개(Merchant Brokering)단계에서는 이전 단계의 고려사항들을 결합하여 ‘누구에게서 구매할 것인가’에 대한 결정을 도와주는 단계로서 상점/상인 관련정보를 사용한다. 이 단계에서는 소비자가 제시하는 기준(예: 가격, 제품보증, 유용성, 배송시간, 명성 등)에 기초하여 선택대상 상인(어느 상인에게서 사야할지)을 평가한다. 협상(negotiation)단계의 핵심은 거래조건(term)을 어떻게 결정할 것인가이다. 전통적 소매 시장(예: 백화점, 서점 등)에서는 제품가격 등에 관해 협상을 위한 여지가 없이 고정되는 경우가 많다. 그 외의 시장(예: 주식, 자동차, 예술품, 재

† 본연구는 2000학년도 경기대학교 학술연구비(일반연구 과제) 지원에 의하여 수행되었음.

\* 경기대학교 경영학부

래시장 등)에서 가격의 협상이나 다른 요소에 대한 흥정이 주요 요소로 작용하기도 한다[9].

## 2.2. 협상결정함수

일반적으로 소프트웨어 에이전트들이 다른 에이전트들과의 협상의 수단으로 사용하는 협상결정함수들은 다음과 같다[5]. 1) 어떤 초기 제안을 보낼 것인가, 2) 합의가 가능한 협상범위는 어느 정도인가, 3) 어떤 역 제안(count offer)을 할 것인가, 4) 어느 시기에 협상을 포기해야 할 것인가, 5) 언제 합의에 도달할 것인가.

초기 제안과 허용 가능한 합의점의 범위는 사용자가 제공하는 값이다. 제안  $i$  를 위하여 에이전트에게 받아들여 질만한 가치의 범위  $a$  는 조건범위  $[minai, maxai]$ 로 표현될 수 있다. 사용자가 에이전트에게 할당하는 최저가치는  $minai$ 로 나타내며,  $maxai$ 는 최대가치를 뜻한다. 이때 구매 에이전트  $b$  가 특정시간  $t$  에 제안  $i$  을 위해 판매 에이전트  $s$  로부터 제안  $O_{ts} \rightarrow b(i)$  를 받는다면, 가치점수 평가함수  $score(O_{ts} \rightarrow b(i))$  를 사용하여 제안을 평가한다. 만약 제안의 가치가 시간  $t+1$  에 그 에이전트  $b$  가 보낼 준비를 하고 있는 역 제안 가치보다 높을 경우에는, 에이전트  $b$  는 그 제안을 수락한다. 그렇지 않은 경우에는, 에이전트  $b$  는 그 제안을 거부하거나 역 제안을 선택한다. 이러한 결정 프로세스를 수리적으로 표현하면, 구매 에이전트  $b$  와 그에 연관된 가치점수 평가함수  $score(O_{ts} \rightarrow b(i))$  를 가지고, 시간  $t$  에 판매 에이전트  $s$  에게 보내진 시간  $t+1$  에서의 제안  $O_{ts} \rightarrow b(i)$  에 대한 에이전트  $b$  의 해석은 다음과 같이 정의된다.

$$I(t+1, O_{ts} \rightarrow b(i)) = \begin{cases} \text{reject} & \text{if } t+1 > t_{max}, \\ \text{accept} & \text{if } score(O_{ts} \rightarrow b(i)) \geq score(O_{ts} \rightarrow b(i)), \\ O_{ts} & \text{otherwise.} \end{cases}$$

$t_{max}$  는 에이전트  $b$  가 협상을 완료해야만 하는 시간을 나타내는 상수로서 협상 에이전트들 간에 시간 한계를 초과하면 협상은 결렬된다. 이 시간 내에 한 에이전트가 다른 에이전트의 제안을 수락한다면 협상은 성사된다.

에이전트가 역 제안을 준비하기 위하여 각 변수의 새로운 가치를 생성하는 몇 가지 협상 결정 함수를

사용할 수 있다. Faratin 등은 3가지 타입의 협상 결정함수(시간 의존적/자원 의존적/행동 의존적 협상 결정함수)를 제안했다[5].

### 2.2.1. 시간 의존적 결정함수

시간 의존적 결정함수에서는 다음 제안을 위한 가치를 결정하기 위해 시간을 유일한 요소로 고려한다. 따라서 이 함수는 남아있는 협상시간에 따라 다음 제안을 위해 수용 가치의 변화로 구성되어 있다[5].

Kasbah[2]는 제품 거래를 위해 사용자가 구매 에이전트와 판매 에이전트를 생성하고 에이전트들 간에 협상을 진행하는 웹 기반의 멀티-에이전트 시스템이다. 제품을 구매하거나 판매하기 위해서, 사용자는 하나의 에이전트를 생성하여 에이전트에게 몇 가지의 구매 및 판매를 위한 전략적 지침을 내리고 협상에 임하도록 한다. 에이전트는 사용자에게서 받은 지침을 바탕으로 사용자 편에서 잠재적인 구매자나 판매자를 탐색하고 그들과 협상한다. 각 에이전트들의 목표는 사용자들에 의한 주문사항(가격협상에서 가능한 최대 및 최소가격, 거래가 진행/마감되는 시점)을 적용하여 구매자 또는 판매자가 수락할만한 거래협상을 완수하는데 있다. 협상이 진행되면서 1, 2, 3차원 함수를 적용하여 제안가격을 수정하는 협상전략을 사용한다. 1차원 감소함수는 판매 에이전트는 요구가격을 시간의 변화에 따라 일정 비율로 낮추어 나간다. 2차원 감소함수는 협상 시간이 거의 종료될 때까지 1차원 감소함수를 사용한 것보다 더 높은 가격을 유지하다가 협상종료시간에 임박하여 가능한 최소가격으로 낮춘다. 3차원 감소함수는 2차원 감소함수와 유사하지만 초기 값을 보다 오랫동안 유지한다. 그러나 자동협상 능력에도 불구하고 Kasbah 에이전트는 가격요소만을 고려하는 단순협상전략을 사용하고 기타 환경을 고려하지 않아 실질적인 적용에 문제점을 갖고 있다.[14]

### 2.2.2. 자원 의존적 결정 함수

자원 의존적 결정함수의 개념은 에이전트가 많은 자원을 사용할수록 협상과정에서 양보를 해야 할 부담이 줄어든다는 것이다. 자원 의존적 결정함수는 시간 의존적 함수와 유사하지만 시간자원은 협상진행 과정에서 지속적으로 감소하는 반면, 다른 자원들은

활용에 있어서 다른 형태를 보이기도 한다.

자원 평가 전략은 특정 자원을 어떻게 소비하는가를 평가하고 이에 따라 역 제안을 생성한다. 협상 중인 에이전트의 수를 자원으로 생각한다면 많은 에이전트가 협상할수록 합의를 위한 부담이 적게 요구될 것이고 시간을 자원으로 모형화하면 협상 중인 에이전트가 많을수록 합의도출을 위한 노력이 적게 필요하고, 협상시간이 길어질수록 부담은 늘어난다[5].

### 2.2.3. 행위 의존적 결정 함수

이 협상전략들은 협력적 문제해결 협상 환경에 있어서 중요하다는 것으로 협상 상대자의 이전 행위에 기초하여 다음에 제시할 제안을 설계한다[4]. 다양한 행위 의존적 결정함수들은 이전 행위를 모방하는 방법에 따라 구분된다. 비례적 또는 절대적으로 모방하거나 이전 몇 가지 제안의 비례적 평균을 계산하여 모방을 결정한다[5].

## 3. 경쟁적 환경을 위한 협상모형

전자상거래와 같은 경쟁적 환경에서 소프트웨어 에이전트의 협상 모델을 현실화하기 위해서는 협상 규약(negotiation protocol), 거래 프로세스(dealing process), 협상 결정함수(negotiation decision function)를 정의해야 한다.

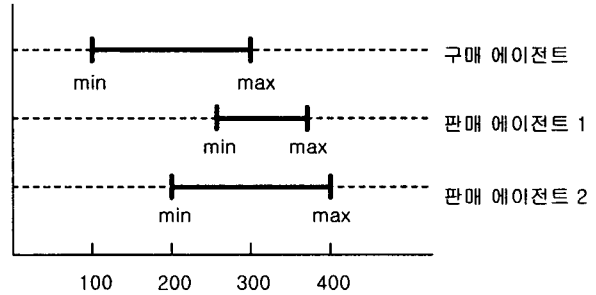
### 3.1. 협상 규약

경쟁 환경에서 소프트웨어 에이전트를 위한 협상 규약은 동시다발적인 협상이 가능해야 하며 또한 여러 결정요소를 고려해야 한다.

#### 3.1.1. 다차원 협상

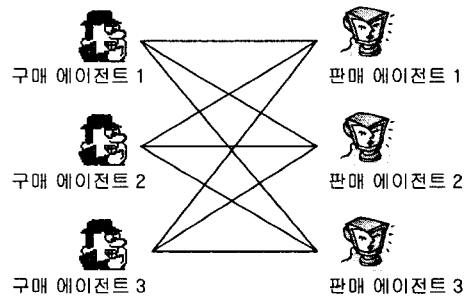
많은 구매자 및 판매자들이 공통된 관심사를 가지고 있는 경쟁적 상황에서, 만약 구매 에이전트가 시간과 공간의 제약 속에서 사람들이 협상하듯이 순차적 협상방식을 사용한다면, 궁극적으로 다른 구매 에이전트보다 좋지 않은 조건으로 계약을 맺는 일이 일어날 것이다. 예를 들어, 그림 1에서, 구매 에이전트가 어떤 제품을 구매하기 위해 협상을 시작하는 경우에 판매

에이전트 2의 초기 판매가가 판매 에이전트 1에 비해 낮기 때문에 첫 번째 협상대상자로 선택하여 협상을 진행한다면 결과적으로는 판매 에이전트 1이 더 나은 조건을 갖고 있음에도 불구하고 판매 에이전트 2와 계약을 체결하는 상황이 일어날 수도 있다.



[그림 1] 순차적 양자 협상의 한계

하지만 그림 2와 같이 많은 경쟁자와 협상 상대자가 공존하는 경쟁적 환경에서 공간 제약 없는 소프트웨어 에이전트는 동시다발적 다차원 협상 접근법을 택해야 한다. 그리고 그들은 새로이 시스템에 진입하는 소프트웨어 에이전트가 더 나은 조건을 갖고 있음에도 불구하고 시장에 진입하는 시간의 차이 때문에 협상에서 배제되는 것을 방지하기 위해서 진행되고 있는 협상에 참여할 수 있는 개방적 협상 규약을 택해야 한다.



[그림 2] 동시다발적 다차원 협상

#### 3.1.2. 다중 결정요소 협상

인터넷 쇼핑몰에서 흔히 사용하는 가격 비교 에이전트는 구매자(또는 판매자)들의 의사결정을 도와주는데 단지 가격요소만을 고려한다는 단점을 가지고 있

었다. 사용자들은 종종 다양한 결정기준에 따라서 제품을 선택하기 때문에, 가격 이외에도 다른 협상결정요소를 고려해야 할 것이다. 따라서 소프트웨어 에이전트들 간에도 가격뿐만 아니라 다른 기준도 협상프로세스에 포함시켜야 한다.

다중 결정요소 협상을 위한 변수들은 아래와 같이 정의될 수 있다.

$$a \in \{b, s\}$$

$$i \in \{1, 2, K, n\}$$

$$weight_i^a \in [0, 1] \text{ where } \sum_{i=1}^n weight_i^a = 1$$

$$O_{i \rightarrow a}^a(i) \in [\min_i^a, \max_i^a]$$

$$O_{i \leftarrow a}^a(i) \in [\min_i^a, \max_i^a]$$

$$score^a(O_{i \rightarrow a}^a(i)) \in [0, 1]$$

$$score^a(O_{i \leftarrow a}^a(i)) \in [0, 1]$$

$$E^a(s) = \sum_{i=1}^n weight_i^a \times score(O_{i \rightarrow a}^a(i))$$

$$E^{a1}(b) = \sum_{i=1}^n weight_i^a \times score(O_{i \leftarrow a}^a(i))$$

간단한 예를 생각해 보면, 판매 에이전트의 협상평가 요소는 '가격'과 '배송시간'을 포함하고 있고 또한, 구매 에이전트는 가격과 배송시간에 관한 조건 값으로 가격에 관해서는 [10, 20]을 배송시간에 대해서는 [1, 5]를 갖고 있다. 또한, 구매 에이전트가 가격요소에 더 높은 비중을 할당함에 따라 배송시간보다 가격을 더욱 더 중요하게 생각한다고 가정하자. 즉,  $[weight_{price}, weight_{delivery\ time}] = [0.9, 0.1]$ . 그렇다면 가격 평가점수는 아래와 같이 1차원 함수로서 모형화 할 수 있을 것이다.

$$score^a(O_{i \rightarrow a}^a(price)) = 1 - \frac{O_{i \rightarrow a}^a(price) - \min_{price}^a}{\max_{price}^a - \min_{price}^a}$$

어느 한 시점에서의 두 판매 에이전트가 제안한 두 개의 협상제안, [가격, 배송시간] = [15, 3]과 [17, 5]에 대해 위 함수를 사용하여 평가점수를 계산한다면, 첫 번째 제안의 가격에 대한 점수는  $1 - [(15-10)/(20-10)] = 0.5$

이고 배송시간에 대한 점수는  $1 - [(3-1)/(5-1)] = 0.5$ 가 된다. 최종 제안 평가점수는 각각의 이슈에 대한 점수에 가중치를 적용하여 합한 것이 된다. 즉,  $0.9 \times 0.5 + 0.1 \times 0.5 = 0.5$ 가 된다. 같은 방법으로, 두 번째 판매 에이전트의 제안의 평가점수는 0.37로 계산될 수 있다. 논리적 행동은 효용가치를 극대화 시켜준다는 기본적인 가정 아래, 구매 에이전트는 첫 번째 판매 에이전트를 유력한 협상상대로서 고려하게 될 것이다.

### 3.2. 환경 의존적 결정 함수

시간 의존적 결정 함수는 협상환경 변화에 대한 고려 없이 단지 시간의 경과에 따른 대응으로 역제안을 생성한다. 하지만, 자원 의존적 결정 함수는 협상시간과 같은 자원을 고려하여 역제안을 생성한다. 따라서 자원 의존적 결정함수는 시간 의존적 함수에 비해 보다 유동적이다.

Faratin 등은 결정함수의 실험을 위해 British Telecom의 네트워크 라인 임대사업에 적용하였다[5]. 하지만 판매 에이전트의 입장에서 자원 의존적 결정 함수를 수정작업 없이 경쟁적 시장에 적용함으로 몇 가지 문제점을 야기하였다. 문제점으로 첫째, 유동적 기한제한 결정 함수를 위해 협상 자원의 양에 따라서 에이전트는 자유롭게  $t_{amax}$ 를 조정할 수 있었기 때문에 판매자가 신속하게 판매하고자 할 때에도 판매 에이전트가  $t_{smax}$ 를 무한대로 늘림으로 인하여 협상을 놓쳐버릴 수도 있다. 둘째, 제한시간동안 얼마나 많은 구매 에이전트들이 하나의 판매 에이전트와 계약을 맺느냐에 중점을 두고 있기 때문에 결정함수가 경쟁적 상태를 반영하지 않는다. 따라서 이러한 함수는 판매 에이전트가 구매 에이전트보다 더 유리한 불공평한 거래를 발생시킬 수도 있었다.

본 연구의 적용 영역과 같은 경쟁적 비즈니스 환경에서 거래의 공정성을 보장하기 위해서는 협상 상대자들뿐만 아니라 경쟁자들의 영향요소도 협상 결정함수에 포함되어야 한다. 따라서 본 연구에서는 협상 당사자들의 수와 협상시간뿐만 아니라 경쟁자들의 수를 반영하여 경쟁적 환경을 기반으로 한 협상 결정 함수를 제안한다.

(i) 구매 에이전트 쪽에서

$$O_{\text{min}}(t) = \min + \alpha^b(t) \times (\max - \min),$$

where 
$$\alpha_i^b(t) = \frac{e^{\left[ \frac{\min + \max}{t \max} \right]^{\frac{1}{k}} - 1}}{e - 1}$$

(ii) 판매 에이전트 쪽에서

$$O_{\text{max}}(t) = \max - \alpha^s(t) \times (\max - \min),$$

where 
$$\alpha_i^s(t) = \frac{e^{\left[ \frac{\min + \max}{t \max} \right]^{\frac{1}{k}} - 1}}{e - 1}$$

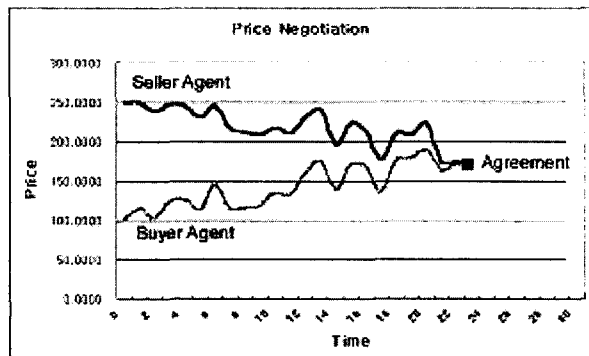
$\alpha(i)$ : 결정요소  $i$  에 관한 다음 제안의 값

$ns(t)$ : 시점  $t$  에서의 판매자의 수

$nb(t)$ : 시점  $t$  에서의 구매자의 수

$e = 2.718282$

초기 제안값은 구매 에이전트를 위해서는 하한가가 되어야 하고, 판매 에이전트를 위해서는 상한가가 되어야 하기 때문에 이 함수에서는 시간 의존적 결정 함수의  $k$  는 배제하였다. 제안값이 사용자가 할당한 조건 값으로부터 무조건적으로 증가하거나 감소되는 것은 합리적이지 못하다.  $nb(t)$  와  $ns(t)$  구매 에이전트와 판매 에이전트의 수를 나타낸다. 만일 시장에 많은 판매 에이전트가 있다면 구매 에이전트는 요구 가격을 천천히 증가시키는 반면, 동일한 상품을 구매하기를 원하는 구매 에이전트가 아주 많다면 판매 에이전트는 요구가격을 신속하게 올릴 것이다. 판매 에이전트에게 있어서는 역이 성립된다.



[그림 3] 환경 의존적 결정 함수를 사용한 가격 협상

그림 3은 환경 의존적 결정함수를 사용하여 구매 에이전트와 판매 에이전트간의 협상 프로세스의 형태를 보여주고 있다. 시간 의존적 결정함수를 사용한 협상 진행 그래프가 규칙적인 모양으로 진행한데 반해 그림 3에서 볼 수 있듯이 환경 의존적 결정함수를 사용할 경우 매번 제안이 생성되는 시장에서의 동적 경쟁에 영향을 받아 불규칙적인 형태를 보여주고 있다.

#### 4. 협상 결정 함수의 실험적 평가

몇 가지 협상 결정함수에 대해 소개하였지만 이론적인 모형 만으로서는 경쟁적 협상 상황에서 어떤 결정 함수가 성공적일 지에 대해 예상할 수는 없다. 따라서 경쟁적 환경을 시뮬레이션하고 시간의존적 결정 함수와 환경의존적 결정함수를 측정하여 성능평가를 통한 차이점을 실험하였다.

##### 4.1. 협상환경과 결정 함수

에이전트의 수와 협상에서 논의되는 결정요소, 언제 합의가 도출되어야 하는지에 대한 만료기한, 에이전트의 목표 등이 협상환경을 결정한다. 무한히 많은 잠재적 환경이 존재하기 때문에, 본 연구에서는 에이전트의 협상 성능을 평가할 수 있는 대표적인 환경을 선택하여 시뮬레이션하였다. 즉 가상의 한 가지 상품을 대상으로 여러 판매자와 구매자가 경쟁을 하는 환경이다.

결정요소를 가격으로 제한할 때 실험적 협상환경은 다음과 같은 변수에 의해 정의된다.

$$k, \beta, t_{\max}, \min_{price}^a, \max_{price}^a, ns(t), nb(t)$$

$k$  는 시간 의존적 결정함수를 검사하는데 사용되는데 Faratin[5] 등이 실험에서 사용한 것과 동일한 값, 0.1로 설정된다. 시간 의존적 결정 변수를 평가하는데 사용되는  $\beta$  값은 0.7, 0.8, 0.9, 1.0 내에서 무작위로 추출된다. 이는 Faratin[5]의 실험에서 가장 성공적인 성과를 얻은 값이다. 는 협상하는 모든 구매/판매 에이전트들을 위해 10~50초 내에서 무작위로 추출된다. 무작위 숫자를 사용하여 협상범위(가격에 대한 에이전트의 최소값과 최대값의 차이)를 계산한다. 구매 에이

전트가 최소가격으로 \$150을, 최대가격으로 \$200을 가지며 판매 에이전트의 최소가격은 \$125와 \$200 사이에서 무작위로 선택되고 최대가격은 \$150과 \$225 사이에서 무작위로 선택된다.  $\beta$  와 비교하기 위해서, 구매 에이전트의 수와 판매 에이전트의 수는 7과 10 사이에서 무작위로 선택된다.

Faratin의 실험은 어떤 결정함수가 최대한의 시간 한도 내에서 대부분의 거래를 성사시키는지에 대해 평가하는 것인 반면에, 본 실험은 경쟁적 상황에서 어떤 결정 함수가 협상 에이전트들 간에서 가장 적절한 이익을 보장하는지에 대해 평가하는 것이다. 이전 실험에서 가장 성공적인 결정 함수는  $\beta$  값이 0.7에서 1.0 사이인 시간 의존적 결정함수이고, 그 다음은 가 5, 10내이고  $\kappa=1$  인 자원 의존적 결정함수였다. 따라서 본 논문에서는 두 개의 결정 함수, 즉 가장 성공적인 성과를 보여준 시간 의존적 함수와 본 논문이 제안하는 환경 의존적 함수의 성과를 비교한다.

## 4.2. 실험적 측정기준

결정함수의 효과성을 평가하기 위하여 결정함수가 에이전트에게 주는 내재적 이윤과 내재적 효용을 완벽정보게임과 비교했을 때의 성과 측정이 필요하다.

### 4.2.1. 내재적 이윤

내재적 이윤은 협상 결과를 최악의 협상사례와 비교했을 때 에이전트가 얻는 이윤으로 나타낸다. 내재 이윤은 주어진 환경에서 소모한 시간과 자원과는 독립적인 협상 최종 결과에 대한 에이전트의 효용성으로서 모형화할 수 있다[7]. 협상결과  $x$  에 관한 구매 에이전트 효용성  $U_{buyer}(x)$  와 판매 에이전트 효용성  $U_{seller}(x)$  는 아래와 같이 1차 함수를 사용하여 계산한다.

$$U_{buyer}(x) = \frac{\max_{price}^b - x}{\max_{price}^b - \min_{price}^b},$$

또는

$$U_{seller}(x) = \frac{x - \max_{price}^s}{\max_{price}^s - \min_{price}^s}$$

만약 협상에 의해 아무런 계약이 성사되지 않았다면  $U_{buyer}(x)$  와  $U_{seller}(x)$  에 0을 할당한다.

### 4.2.2. 획득 효용성

내재적 효용성은 에이전트의 주관적인 성과를 측정한다. 하지만 결정함수의 공정성을 측정하기 위해서는 제어조건이 필요하다. 완벽한 정보를 가진 판매자 및 구매 에이전트를 위한 이러한 제어환경  $C$  는 다음과 같이 정의할 수 있다[5].

$$C = \frac{\max_{price}^s + \min_{price}^b}{2}$$

전 절에서 보았던 효용성에 대한 정의를 적용하면 에이전트  $a$  를 위한 완벽 정보게임의 효용성  $Ua(C)$  는 계산될 수 있다. 이것으로 에이전트의 비교 성과, 즉 획득 효용성  $GUa$  는 내재적 효용성과 완벽정보 상황에서 에이전트가 얻을 효용성과의 차이를 사용하여 아래와 같이 계산된다[5].

$$GU_a = U_a(x) - U_a(C)$$

## 4.3. 실험적 평가결과

두 개 함수의 이론적인 기초를 기반으로 각 함수의 성능에 대한 두 가지의 평가를 아래와 같이 예측할 수 있다.

예측 1:

환경 의존적 결정함수는 유동적 시장 상황에 따라 협상 가격이 유동적으로 불규칙적인 변화를 보이기 때문에 시간 의존적 결정함수보다 보다 많은 협상 시간을 소비할 것이다.

예측 2:

환경 의존적 결정함수가 시간 의존적 결정함수보다 더욱 많은 협상 시간을 소비하지만 환경 의존적 결정함수는 유리한 조건을 갖고 있는 에이전트가 획득하는 이익을 뚜렷하게 반영하기 때문에 시간 의존적 결

정 함수보다 더 나은 내재적 효용성과 획득 효용성을 가질 것이다.

본 평가의 목적이 일대일 협상 결과에서 평가된 내재적 효용성과 획득 효용성을 계산하는 것이기 때문에 실험에서 사용한 협상은 양자 협상규약에 기초한다. 그리고 표본평균의 표준편차가 0.01 보다 클 가능성을 0.05 이하로 하기 위해서 각 협상 결정함수 당 1,000개의 협상을 시뮬레이션하였다.

표 1은 거래가 성사된 사례에서 경과된 시간의 평균과 표준편차를 보여준다.

[표 1] 경과 시간에 대한 평균과 표준편차

	시간 의존적 결정함수	환경 의존적 결정함수
평균 (초)	15.848	18.761
표준편차	3.678	4.199

예상대로 환경 의존적 결정함수가 시간 의존적 결정함수보다 많은 협상시간을 소비한다는 것을 실험을 통해서 알 수 있다. 보다 많은 시간소비의 원인으로는 환경 의존적 함수가 동적인 경쟁적 시장환경을 고려함으로 불규칙적으로 진행된다는 점과 시간 의존적 함수에서의  $k$  는 합의에 이르기 위해 양쪽 에이전트를 격려한다는 특성 때문이라고 생각할 수 있다.

표 2는 거래가 이루어지진 사례의 모든 협상에 대한 내재 효용성과 획득 효용성의 평균 및 표준편차를 보여준다.

[표 2] 내재 효용성과 획득 유용성

	시간의존적결정함수		환경의존적결정함수	
	내재효용성	획득효용성	내재효용성	획득효용성
평균	0.496	0.044	0.533	0.088
표준편차	0.123	0.23	0.139	0.36

실험결과는 경쟁적 상황에 대한 고려가 없는 시간 의존적 함수는 환경 의존적 함수보다 더 적은 편차를 가지고 있을 것이라는 예상을 뒷받침한다. 이는 시간 의존적 함수가 규칙적인 유형으로 움직이는 반면에 환경 의존적 함수는 불규칙적으로 진행된다는 특징에 의한 것이라고 생각되어진다. 하지만 환경 의존적 결정함수를 사용하는

구매 에이전트나 판매 에이전트가 유리한 상황이 될 때, 시간 의존적 결정함수를 사용하는 에이전트보다 더 많은 내재 효용성과 획득 효용성을 얻게 된다.

## 5. 결론

본 논문에서는 경쟁 비즈니스 환경 상에서 운영상의 이득과 협상 결정함수의 단점을 평가하기 위해서 다양한 협상 결정함수의 실험적 평가를 시도했다. 실험결과는 특정 협상 결정 함수, 즉 환경 의존적 결정 함수의 상대적인 장점을 명확하게 보여준다. 대부분 에이전트 시스템들이 그들의 협상 결정함수를 완전히 공개하지 않았기 때문에 더 많은 협상 결정함수들을 평가할 수 없었다. 그럼에도 불구하고 본 연구는 향후 현실적인 전자상거래 에이전트 시스템 설계에 있어 어떻게 협상 의사결정함수를 적용해야 하는 지에 대한 시사점을 제공한다.

결정함수 연구에 있어서 특정 환경에서 가중치를 사용한 협상 결정함수 조합은 순수한 결정함수보다 더 나은 협상을 수행할 것이라는 주장과 에이전트가 주어진 환경에 따라 어떻게 가중치를 역동적으로 변화시킬 것인가에 대한 연구의 필요성을 언급했다. 본 논문에서 제안된 소프트웨어 에이전트 역시 다른 에이전트들과 협상하기 위하여 하나의 결정 함수를 사용하고 있고 향후 본 연구는 다음과 같은 방향으로 확장하려고 한다. 협상 전략의 효과에 대해 보다 면밀한 조사가 필요하다. 또한 에이전트에 더 많은 활동성을 추가하여 에이전트가 결정 함수의 가중 조합을 어떻게 환경에 따라 동적이고 지능적으로 변화할 수 있는지에 대해 연구가 진행 중이다.

## 참고 문헌

- [1] Kalakota, R. and A. B. Whinston, *Frontiers of Electronic Commerce*, Addison-Wesley, 1996.
- [2] Kasbah, URL: <http://agents.www.media.mit.edu/groups/agents>
- [3] Pruitt, D. G., *Negotiation Behavior*, Academic Press, 1981.
- [4] Axelrod, R., *The Evolution of Cooperation*,



Basic Books, 1984.

- [5] Faratin, P., C. Sierra, and N. R. Jennings, Negotiation decision functions for autonomous agents, *Robotics and Autonomous Systems*. *Int. Journal of Robotics and Autonomous Systems*, 24(3-4) (1998), 159-182.
- [6] Raiffa, H., *The Art and Science of Negotiation*, Harvard University Press, 1982.
- [7] Russell, S., and E. Wefald, *Do the right thing*, MIT Press, 1991.
- [8] Lee, K.J., Y.S. Chang, and J.K. Lee, Time-bound negotiation framework for electronic commerce agents, *Decision Support Systems* 28 (2000), 319-331
- [9] Guttman, R., A. Moukas, and P. Maes, Agent-mediated Electronic Commerce: A Survey, *Knowledge Engineering Review*, 13(2) (1998), 147-159.
- [10] BargainFinder, URL: <http://bf.cstar.ac.com/bf>
- [11] Jango, URL: <http://www.jango.com>
- [12] Doorenbos, R., O. Etzioni, and D. Weld, A Scalable Comparison-Shopping Agent for the World Wide Web, *Proceedings of the First International Conference on Autonomous Agents*, 1997.
- [13] Guttman, R., and P. Maes, Agent-mediated Integrative Negotiation for Retail Electronic Commerce, *Proceedings of the Workshop on Agent Mediated Electronic Trading*, 1998.
- [14] Kang, J.Y., and E.S. Lee, A Negotiation Model in Electronic Commerce to Reflect Multiple Transaction Factors and Learning, *Proceedings of the Twelfth International Conference on Information Networking*, 1998.



김 중 한 (Joong-Han Kim)

1985 고려대학교 수학과

1988 Bowling Green State

Univ. 전산과학 석사

1993 University of Nebraska -

Lincoln 경영정보학 박사

1996 ~ 현재 경기대학교 경영학부 부교수

관심분야 : 전자상거래, 정보통신기술의 전략적 활용,  
SI 정책