

기업간 전자상거래에 있어서 협상의사결정지원을 위한 정수행렬 연산 추론 메커니즘에 관한 연구*

이 건 창**, 조 형 래***, 권 순 재****

An Integer Matrix-Driven Inference Mechanism for Negotiation Decision Support in B2B Electronic Commerce

Lee, Kun Chang, Cho, Hyung Rae, Kwon, Soon Jae

This paper is aimed at proposing a new inference mechanism for B-to-B electronic commerce decision-makings, named IMITAO (Integer Matrix-driven Inference based on Transformed And-Or graph) which is based on integer matrix operation to speed up the inference. During the B-to-B electronic commerce, many kinds of negotiations are needed for mutually satisfactory decision-makings. During such negotiations, several factors including subjective and objective constraints should be considered so as to reach satisfactory decisions. In this respect, we suggest first a Transformed AND-OR Graph (TAOG) which each firm's conditions or judgement are incorporated into, and then we propose a high-speedy inference mechanism named IMITAO which basically depends on TAOG. Firms engaged in B-to-B negotiations on the Internet can get an appropriate decision support from using IMITAO for their negotiation simulations. The proposed IMITAO inference mechanism is characterized by its fast inference because its main inference procedures are based on integer matrix operation. A real-world example for B-to-B negotiations was used to prove the validity of our proposed TAOG and IMITAO approach. Experimental results showed that our approach was very useful in performing the B-to-B electronic commerce decision-makings considering a wide variety of either subjective or objective constraints.

* 본 논문은 1999년도 성균관대학교 교내 삼성학술연구비 지원을 받아 수행된 것임.

** 성균관대학교 경영학부 교수

*** 경상대학교 산업시스템공학부 교수

**** 성균관대학교 경영학부 박사과정

I. 서론

인터넷을 이용한 전자상거래는 시간적, 공간적 한계를 극복할 수 있고, 전세계를 실시간으로 연결하여 하나의 거대한 가상시장을 제공할 수가 있기 때문에, 향후 중요한 비즈니스의 대상으로 떠오르고 있다[O'Connor & O'Keefe, 1997; Westland and Clark, 1999]. 특히 최근 전세계적으로 웹사이트의 수가 10억여개로 증가하고 인터넷 사용자가 2억명에 이르게 되자 전자상거래의 유망성이 더욱 두드러지게 되었다. 특히 기존의 전통적 패러다임에 의하여 지배되던 기업간 거래는 빠른 속도로 전자상거래로 대체되고 있는 실정이다[Shaw et al., 1997]. 그러나, 아직 많은 기업들이 전자상거래가 가지고 있는 전략적 의미와 기술적 응용가능성에 대하여 제대로 인식하지 못한 상태에서, 단순히 인터넷상에 웹사이트를 만들기만 하면 전자상거래가 잘 될 수 있다는 단순한 사고방식을 가지고 있는 것이 현실이다[Rebello et al., 1996].

그렇다면 본 논문의 연구의도를 보다 정확하게 소개하기 위하여, 전자상거래에 관한 기존연구를 간략히 정리하여 보자. 우선, 전자상거래에 관한 연구에는 전자상거래의 전략적 활용과 정보기술과의 연관성을 조사한 연구를 비롯하여 [Lederer et al., 1996, 1997], 인터넷상에서의 소비자 행동에 관한 연구[Hoffman & Novak, 1996; Jarvenpaa & Todd, 1997; Kim, 1997; Dennis, 1998], 인터넷 쇼핑물 유형 및 디자인에 관한 연구[Lohse & Spiller, 1998; Spiller & Lohse, 1998; O'Keefe & Mceachern, 1998], 지능형 에이전트 활용에 관한 연구[Chavez & Maes, 1996; Schrooten, 1996] 등이 있다.

그러나, 전자상거래에 관한 기존연구를 살펴보면 전자상거래 문제를 복잡한 의사결정문제로 보고, 이 같은 문제해결을 위한 객관적인 요인들뿐만 아니라 주관적인 요인들도 함께 고려하

여 보다 합리적이고 지능적인 의사결정을 지원하는 방법론에 관한 연구는 거의 없음을 알 수 있다. 특히 기업간 전자상거래 문제의 경우는 대개의 경우 두 거래당사자 기업이 인터넷 공간에서 최적의 거래조건 타결을 위하여 협상을 하여야 한다. 따라서, 기업간 전자상거래는 개인간 전자상거래보다도 고려하여야 할 요인들이 매우 많다. 예를 들어서 기업간에 서로 협상의 대상이 되는 가격, 주문수량, 지불조건, 납기일, 배달조건, 품질 등과 같은 소위 객관적인 요인을 포함하여, 노조와의 관계, 원자재 수급조건 등 다양한 주관적인 요인들도 있기 마련이다. 이러한 경우 해당 기업이 가지고 있는 여러 가지 주관적 요인을 동시에 반영할 수 있는 추론메커니즘이 절대적으로 필요하다.

따라서, 본 연구에서는 이러한 기존연구의 한계를 극복하고자 하는 시도에서 기업간 전자상거래 문제를 전제로 하여

- (1) 다양한 객관적, 그리고 주관적 요인을 효과적으로 반영하여 기업간 전자상거래 협상의 사결정 문제해결을 지원할 수 있는 추론 메커니즘을 개발하고,
- (2) 해당 추론 메커니즘의 성과를 측정하기 위하여 실제 기업간 전자상거래 문제에 적용하여 그 성과를 분석하고자 한다.

우선, 첫 번째 목적을 위하여 본 연구에서는 두단계를 거쳐서 기업간 전자상거래 협상의사결정지원을 위한 추론 메커니즘을 개발한다. 1단계로서는 기존의 전문가시스템에서 지식표현 기법으로 널리 활용된 AND-OR Graph(Giarratano & Riley, 1994)의 단점을 개선한 TAOG, 즉 "변환된 AND-OR 그래프(Transformed AND-OR Graph)"를 제시한다. 이는 AND-OR Graph에 복합노드 개념을 도입함으로써 복잡한 추론과정을 보다 효과적인 정수행렬 연산방법으로 하기 위한 기초개념이다. 2단계로서는 TAOG에 규칙

행렬과 충분조건벡터 개념을 도입하여 TAOG로 표현된 지식구조를 기초로 한 추론과정을 정수행렬화한다. 이를 본 연구에서는 IMITAO, 즉 “TAOG에 기초한 정수행렬 연산 추론기법”(Integer Matrix-driven Inference based on Transformed AND-OR Graph)로 명명한다. 두 번째 목적을 위하여 본 연구에서는 실제 기업간 전자상거래 문제를 기초로 한 문제를 가지고 그 성과를 분석한다. 이 문제는 우리나라에서 기업간 전자상거래 문제를 가장 널리 다루는 공식기관 중의 하나인 KTNET(주)에서 실제로 사용된 사례를 기초로 한 것이다.

본 연구에서 제안하는 TAOG와 IMITAO에 대한 자세한 내용은 2장과 3장에서 보다 상세하게 설명이 되겠지만, 가장 큰 특징은 기존의 AND-OR 그래프가 가지고 있는 지식표현 기법을 개선하고(이는 TAOG의 특징임), 여기에 규칙행렬 개념과 충분조건벡터 개념을 도입하므로써 인터넷에서 기업간 전자상거래 협상의사결정 지원에 사용될 수 있는 새로운 추론메커니즘인 IMITAO를 제시하였다는 점이다. 사실 인터넷에서 기업간 전자상거래 관련 의사결정을 지원하려면, 그것도 여러 가지 전략적인 대안을 생각하여야 하는 협상의사결정을 지원하려면 반드시 전문가시스템이 가지고 있는 주관적인 요소 및 지식을 반영하면서도 추론속도가 빨라야 한다. 이러한 목표가 달성되려면

첫째, 전문가시스템의 장점인 다양한 주관적인 요소와 지식반영 능력은 그대로 유지하면서
서도,

둘째, 그 단점인 패턴매칭이라는 느린 추론속도를 개선하는

두가지 노력이 병행되어야 한다. 본 연구에서는 첫 번째 노력을 위하여 기존의 AND-OR 그래프의 지식표현 방법중 단점을 약간 개선한 TAOG를 제안하고, 두 번째 노력을 위하여 규

칙행렬 개념과 충분조건벡터 개념을 도입하여 정수행렬 연산추론이 가능하도록 한 IMITAO를 제시한다.

본 연구의 구성은 다음과 같다. 2장에서는 TAOG, 즉 변환된 AND-OR 그래프 방식의 개념을 소개하고 3장에서는 이를 기초로 한 새로운 개념의 정수행렬연산에 의한 추론메커니즘인 IMITAO를 소개한다. 특히 3장에서는 IMITAO의 이론적 배경을 수학적으로 증명하고자 한다. 4장에서는 TAOG와 IMITAO를 이용하여 실제 기업간 전자상거래 문제를 풀어보고 그 타당성을 분석한다. 5장에서는 본 연구의 결론 및 향후 연구방향을 언급한다.

II. TAOG: 변환된 AND-OR 그래프

본 연구에서 제안하는 TAOG는 이미 전문가시스템에서 기본적인 지식표현 기법 및 추론 메커니즘으로 널리 사용되어온 AND-OR 그래프 방식[Giarratano & Riley, 1994]을 약간 수정한 지식표현 방법이다. 이는 어디까지나 기업간 전자상거래 의사결정지원에 적합한 추론형태를 개발하기 위한 전단계로서의 의미를 갖는다.

기존의 AND-OR 그래프 방식은 전문가시스템에서 기본적인 지식표현 방법 및 추론 메커니즘으로 널리 활용해온 방식이다[Buchanan & Shortliffe, 1984]. 이러한 AND-OR 그래프 방식은 의사결정자의 주관적인 요인을 효과적으로 반영할 수 있다는 장점이 있지만, 한편으로는 패턴매칭에 의하여 추론을 하기 때문에 인터넷에서의 기업간 전자상거래에서와 같이 신속한 추론을 요구하는 영역에는 부적합하다는 단점이 있다. 따라서, 이같은 단점을 극복하면서도 주관적인 요인을 반영할 수 있는 AND-OR 그래프 방식의 장점을 유지하기 위해서 일단 다음과 같이 변환된 AND-OR 그래프 방식을 생각해보자. 우선, AND-OR 그래프 방식은 식 (1)과 같이

IF-THEN 방식의 규칙에 기초하고 있다.

$$\text{IF } \alpha_1 \text{ and } \alpha_2 \text{ and } \dots \text{ THEN } \beta \quad (1)$$

여기서 각 α_i 및 β 는 조건(condition or proposition)을 의미하며, 조건들의 결합(conjunction)인 α_1 and α_2 and ...는 규칙의 전제부분(premise), β 는 규칙의 결론부분(conclusion or action)을 나타낸다. 그리고 규칙의 전제부분이 하나의 조건으로 구성되어 있을 경우 이를 단순규칙(simple rule)이라 하고, 규칙의 전제부분이 둘 이상의 조건으로 결합되어 있을 경우 이를 복합규칙(composite rule)이라 한다. 식 (1)에 나타난 형태의 규칙은 간략히 α_1 and α_2 and ... \Rightarrow β 로 표기된다. 그런데 주어진 영역에서 문제를 해결하기 위한 규칙을 수집하면 항상 식 (1)과 같은 기본 형태를 띠고 있는 것은 아니다. 기본 형태가 아닌 규칙의 유형은 다음과 같다.

- Type 1 : $a \Rightarrow (\beta \text{ and } \gamma)$,
- Type 2 : $(a \text{ or } \beta) \Rightarrow \gamma$,
- Type 3 : $a \text{ and } (\beta \text{ or } \gamma) \Rightarrow \delta$
- Type 4 : $a \Rightarrow (\beta \text{ or } \gamma)$.

하지만 이와 같은 유형의 규칙은 다음과 같이 동등한 기본 형태의 규칙으로 변경할 수 있다. 즉 Type 1은 $a \Rightarrow \beta$ 와 $a \Rightarrow \gamma$ 라는 두 개의 기본 형태 규칙으로 나눌 수 있고, Type 2는 $a \Rightarrow \gamma$ 와 $\beta \Rightarrow \gamma$ 로 나눌 수 있다. 또한 Type 3는 $(a \text{ and } \beta) \Rightarrow \delta$ 와 $(a \text{ and } \gamma) \Rightarrow \delta$ 로 나눌 수 있다. Type 4는 추론에 있어 특정한 의미를 가지지 못하기 때문에 본 연구에서는 고려하지 않는다. 이때 TAOG의 특성을 알아보기 위하여 특정 문제영역에서 수집한 규칙들이 다음과 같다고 하자.

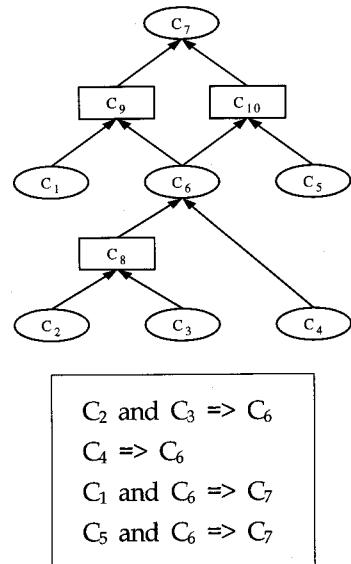
$$C_2 \text{ and } C_3 \Rightarrow C_6$$

$$C_4 \Rightarrow C_6$$

$$C_1 \text{ and } C_6 \Rightarrow C_7$$

$$C_5 \text{ and } C_6 \Rightarrow C_7$$

이를 본 연구에서 제안하는 TAOG로 표현하면 다음 <그림 1>과 같다. <그림 1>에서 보듯이 TAOG는 두가지 형태 (원과 네모)의 노드와 이들을 연결하는 호로 구성된다. 단순노드라 불리는 각 원은 하나의 조건을 나타내고 복합노드라 불리는 각 네모는 조건들의 결합(conjunction)을 나타낸다. 그리고 단순노드로 들어오는 호는 단순 또는 복합규칙을 의미하며 복합노드로 들어오는 호는 결합조건의 구성요소를 나타낸다.



<그림 1> 예제 TAOG

또한 TAOG를 구성하는 노드들은 그 역할에 따라 다음과 같이 여러 가지 종류로 나눌 수 있다.

- 리프노드 : 리프노드(leaf node)란 들어오는 호는 없이 나가는 호만 있는 노드로서 진위 값이 실험 또는 관측을 통해 정해질 수 있는 사실을 의미한다. 예를 들어 <그림 1>에서 리프노드는 C_1, C_2, C_3, C_4 및 C_5 이다.

- **루트노드** : 루트노드(root node)란 들어오는 호만 있고 나가는 호는 없는 노드로서 주로 최종 관심의 대상이 되며 의사결정에 직접적인 영향을 미치는 사실을 나타낸다. 예를 들어 <그림 1>에서 루트노드는 C_7 이다.
- **중간노드** : 들어오는 호 및 나가는 호가 동시에 존재하는 노드로서 리프노드와 루트노드를 연결하는 매개체 역할을 한다. 특히 TAOG에서 복합노드는 항상 중간노드가 된다. 예를 들어 <그림 1>에서 중간노드는 C_6, C_8, C_9 및 C_{10} 이다.
- **전제노드** : 임의의 노드 C_i 에 대한 전제노드란 노드 C_i 로 들어오는 호의 꼬리 부분에 연결되어 있는 노드를 의미한다. 예를 들어 <그림 1>에서 노드 C_6 의 전제노드는 C_4 와 C_8 이다.
- **결과노드** : 임의의 노드 C_i 에 대한 결과노드란 노드 C_i 에서 나가는 호의 머리 부분에 연결되어 있는 노드를 의미한다. 예를 들어 <그림 1>에서 노드 C_6 의 결과노드는 C_9 와 C_{10} 이다.

이상에서 설명한 TAOG와 기존의 AND-OR 그래프와의 가장 커다란 차이점은 복합규칙을 나타내기 위한 복합노드의 도입이라 할 수 있다 (<그림 2> 참조). 그렇다면 복합노드의 의미를 기존의 AND-OR 그래프의 경우와 비교하여 보다 면밀하게 고찰해보자.

첫째, 복합노드란 기존의 AND-OR 그래프에서 사용되는 AND 노드를 단순히 기호와 명칭만을 바꾼 것은 아니다. 오히려 TAOG에서 사용되는 복합노드는 기존의 AND/OR 그래프에서 나타난 AND 노드는 그대로 나타내고, 추론의 효율성 제고만을 위해서 도입된 노드이다.

둘째, 이러한 측면에서 본다면 TAOG에서 적

용되는 복합노드는 기존의 AND/OR 그래프에서 사용되는 AND 노드와 그 의미상 완전히 일치하는 개념은 아니다. 다시말해 TAOG에서 사용되는 복합노드는 복합규칙의 전제부분에 나타나는 조건들의 결합만을 나타내는 Dummy 노드이다.

셋째, 이같이 복합노드는 기본적으로 복합규칙의 전제부분의 충족여부를 정수행렬 연산만으로 효율적으로 판단하기 위해 특별히 도입된 노드개념이다.

이렇게 규칙 또는 지식의 구조를 표현할 때 기존의 AND-OR 그래프처럼 단순노드만을 사용하여 호로 연결시키는 것이 아니라, TAOG에서처럼 복합노드를 도입할 경우의 장점은 다음과 같다.

첫째, 복합규칙들을 구성하는 조건들이 상호 복잡하게 얽혀있을 경우 이를 보다 체계적으로 나타내 보임으로서 규칙의 구조에 대한 이해도를 높이고 잘못된 부분이나 누락된 부분의 발견을 용이하게 한다.

둘째, TAOG를 이용하여 몇몇 수학적 정의를 추가로 하게 되면 이후 설명에서 제시된 것처럼 정수행렬 연산으로만 이루어진 신속한 추론방법인 IMITAO의 개발이 가능하다. 특히 이러한 특징은 기업간 전자상거래의 경우처럼 신속한 추론을 요구하는 의사결정문제에서 특히 요구되는 속성이다.

TAOG가 갖는 첫 번째 장점을 다음과 같은 규칙을 가지고 그림으로 설명해보자.

$$C_1 \text{ and } C_2 \Rightarrow C_4$$

$$C_2 \text{ and } C_3 \Rightarrow C_4$$

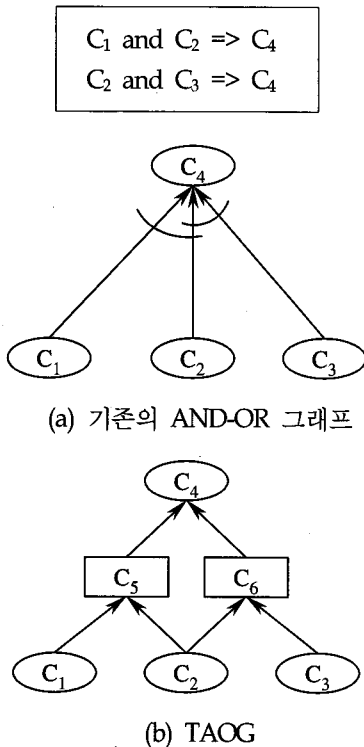
이같은 규칙을 기존의 AND-OR 그래프로 표현하고 TAOG로 각각 표현하면 <그림 2>와 같다. 이때 노드 C_1 과 C_3 만 참(true)일 경우 C_4 는 거짓(false)이어야 한다. 왜냐하면 규칙에

의하면 C_1 과 C_2 모두가 참이거나 또는 C_2 와 C_3 가 모두 참이어야만 C_4 가 참이 되기 때문이다. 그러나, <그림 2-a>에서 보는 바대로 기존의 AND-OR 그래프에서는 노드 C_1 과 C_3 만 참(true)일 경우 노드 C_4 가 거짓이 되어야 함에도 불구하고 참이라고 추론된다는 문제점이 발생한다. 하지만 <그림 2-b>와 같이 TAOG로 이러한 규칙을 표현하게 되면(즉, 지식을 표현하게 되면), C_1 과 C_3 만 참이면 C_5 및 C_6 가 거짓이 되고, 이에 따라 C_4 도 거짓으로 추론된다. 따라서, TAOG에서와 같이 복합노드 개념을 활용하여 주어진 규칙 또는 지식을 표현하게 되면, 해당 복합노드의 전제조건의 거짓 또는 참을 정수로 표현하여 전체적인 추론방법을 기존의 AND-OR 그래프 방식보다 더 체계적이고 정수 행렬 연산화할 수가 있다. 말하자면 추론을 정수연산

화하여 그 효율성을 향상시킬 수 있는 여지가 그만큼 커진다는 것이다. 따라서, TAOG에서는 기존의 AND-OR 그래프에 이같이 복합노드 개념을 도입함으로써 복합규칙의 전제조건의 참 또는 거짓 여부를 숫자로 계산하여 이를 토대로 전문가시스템의 추론을 정수행렬 연산화하는 가능성을 크게 했다는 점에서 의의가 있다. 이는 TAOG에 기초한 추론 메커니즘이(본 연구에서는 IMITAO를 의미) AND-OR 그래프에 기초한 추론 메커니즘보다 효율성이라는 측면에서 큰 진전을 볼 수가 있다는 의미이다.

이같이 IMITAO에 의한 추론의 효율성 증대 부분은 3장에서 보다 구체적인 수학적 정의와 공식을 통하여 확인하도록 하자.

III. IMITAO: 정수행렬에 기초한 추론 메커니즘



<그림 2> 기존의 AND-OR 그래프와 TAOG의 비교

IMITAO는 2장에서 서술한 TAOG를 이용한 추론 메커니즘을 의미한다. 이미 2장에서도 언급한 바와 같이, TAOG는 기존의 AND-OR 그래프에 복합노드를 도입함으로써, 복합규칙의 전제조건의 참 또는 거짓 여부를 정수연산화할 수 있도록 하였다. 따라서, IMITAO는 이같은 TAOG의 특성을 기초로한 정교한 정수연산기법에 의한 추론 메커니즘을 의미한다. 이를 위하여 필요한 정의와 명제를 설명하면 다음과 같다.

[정의 1] 규칙행렬 R: $C_i, i=1, \dots, n$ 을 주어진 TAOG에 나타난 노드들이라 하고, C_i 로부터 C_j 로 가는 호가 있을 경우 이를 $C_i \rightarrow C_j$ 로 나타낸다고 하자. 그러면 앞절에서도 설명하였듯이 $C_i \rightarrow C_j$ 는 C_j 가 단순노드일 경우 $C_i \Rightarrow C_j$ 를, C_j 가 복합노드일 경우 $C_i \in C_j$ 를 의미하게 된다. 이제 n 개의 노드로 이루어진 TAOG에 대한 $n \times n$ 규칙행렬 R은 다음과 같이 정의된다:

$$R_{ij} = \begin{cases} 1, & \text{if } C_i \rightarrow C_j \text{ (즉 } C_i \Rightarrow C_j \text{ 또는 } C_i \in C_j) \\ 0, & \text{otherwise} \end{cases}$$

$i, j = 1, \dots, n$ ■

이렇게 정의된 규칙행렬 R은 주대각선이 모두 1인 특징을 갖는다([$C_i \Rightarrow C_j$] \equiv [$\sim C_i \vee C_j$])로 항상 참이 되므로). 예로서 <그림 1>의 예제 TAOG에 대한 규칙행렬이 <그림 3>에 나타나 있다.

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

<그림 3> <그림 1>의 예제 TAOG에 대한 규칙행렬 R

이러한 규칙행렬에서 주목하여야 할 점은 규칙행렬 R 자체에는 단순노드와 복합노드가 구분되어 있지 않다는 점이다. 따라서 이러한 면에서 본다면 규칙행렬 R은 Adjacent Matrix와 유사하다. 그러나, 이러한 단순노드와 복합노드의 구분, 즉 OR 조건과 AND조건의 구분은 [정의 2]에 나타난 충분조건벡터에 반영되어 있고, 규칙행렬 R에는 반영이 안되어 있다. 따라서 IMITAO에서는 규칙행렬과 충분조건벡터를 결합하여 추론을 전개하여야 한다. 그리고 규칙행렬 R에서 $C_i \rightarrow C_j$ 이면 R_{ij} 값이 1로 되는 것은 추론과정에서 이전단계에서 참으로 밝혀진 노드를 그 다음 추론단계에서도 참이라고 인식하기 위한 것으로서 전적으로 계산상의 효율성을 위한 것이다.

[정의 2] 충분조건 벡터 S : $C_i, i=1, \dots, n$ 을 주어진 TAOG에 나타난 노드들이라 하자. 이 경우 $1 \times n$ 충분조건 벡터 $S = [S_1, \dots, S_n]$

은 다음과 같이 정의된다 :

$$S_i = \begin{cases} 1, & C_i \text{가 단순노드일 경우} \\ NP(C_i), & C_i \text{가 복합노드일 경우} \end{cases}$$

$i = 1, \dots, n$

여기서 $NP(C_i)$ 는 노드 C_i 로 들어오는 호의 수를 의미 ■

이상과 같이 정의된 충분조건 벡터 $S = [S_1, \dots, S_n]$ 의 특성을 살펴보자. 우선 리프노드는 항상 단순노드이므로 임의의 리프노드 C_i 에 대한 S_i 는 항상 1로 정의된다. 그리고 임의의 중간 또는 루트노드 C_i 에 대한 S_i 는 노드 C_i 가 참이 되기 위하여 선행적으로 참이 되어야 하는 전제노드의 수중 최소값을 의미한다. 중간 또는 루트노드 C_i 가 단순노드인 경우 C_i 의 전제노드 중 하나만 참이 되어도 C_i 는 참이 된다. 따라서 임의의 단순노드 C_i 에 대한 S_i 는 항상 1이 된다. 반면에 노드 C_i 가 복합노드인 경우 C_i 의 전제노드 모두가 참이 되어야만 C_i 가 참이 될 수 있다. 따라서 임의의 복합노드 C_i 에 대한 S_i 는 C_i 로 들어오는 호의 수와 일치한다. 예를들어 <그림 1>의 예제 TAOG에 대한 충분조건 벡터 S는 다음과 같다:

$$S = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 2 \ 2 \ 2]$$

[정의 3] 진리상태 벡터 T : TAOG에 나타난 n개의 노드가 갖는 진리상태(truth state)를 T라 하면, 진리상태 벡터 T는 $T = (T_1, \dots, T_n)$ 으로 나타낼 수 있다. 이때, T의 각 성분이 가지는 진리상태는 다음과 같이 정의된다.

$$T_i = \begin{cases} 1, & \text{if } C_i \text{ is true} \\ 0, & \text{if } C_i \text{ is false or unknown} \end{cases}$$

$i = 1, \dots, n$ ■

이때 주어진 리프노드에 대한 진위값에 따라 TAOG에 나타난 노드들 간의 연관관계를 통해 중간 또는 루트노드들의 진위값이 어떻게 변환되는가를 규명하게 되면 추론에 대한 중요한 실마리를 얻을 수가 있다. 이러한 진리상태의 변환(transformation)은 규칙행렬 R과 진리상태벡터 T 및 최소 충분조건 벡터 S를 이용하면 다음과 같이 간단한 정수 연산을 통해 처리할 수 있다. $T^k = (T_1^k, \dots, T_n^k)$ 를 현재의 진리상태라 하고, 벡터 U^k 를 T^k 와 R의 곱 즉,

$$U^k = T^k \cdot R \quad (2)$$

이라 하자. 그러면 U^k 의 각 성분 U_i^k 는 노드 C_i 의 전제노드 중 참인 것의 수를 의미하게 된다. 그리고 $T^{k+1} = (T_1^{k+1}, \dots, T_n^{k+1})$ 을

$$T_i^{k+1} = \begin{cases} 1, & \text{if } U_i^k \geq S_i \\ 0, & \text{otherwise} \end{cases}, \quad i=1, \dots, n \quad (3)$$

이라고 정의하면 T^{k+1} 은 다음과 같은 성질을 갖는다.

[명제 1] 노드 C_i 가 어떤 단순노드 C_j 의 전제노드이고, C_w $h = i^1, \dots, i^p$ 이 어떤 복합노드 C_w 의 모든 전제노드의 집합인 TAOG가 있다고 하자. 그러면 $T_i^k = 1$ 이고, $T_h^k = 1, h = i^1, \dots, i^p$ 인 임의의 진리상태 벡터 T^k 에 대한 T^{k+1} 을 식 (2) 및 식 (3)을 이용하여 구하면 T_i^{k+1} 및 T_w^{k+1} 역시 1이 된다.

[증명] C_i 가 단순노드 C_j 의 전제노드이면 LBS [정의 1] 및 [정의 2]에 의해 $R_{ij} = 1, S_j = 1$ 이 된다. 여기서 $T_i^k = 1$ 이면 식 (2)에 의해

$$U_j^k = \sum_{r=1}^n T_r^k \cdot R_{rj} \geq T_i^k \cdot R_{ij} = 1 \geq S_j (= 1)$$

이 성립하여 식 (3)에 의해 $T_j^{k+1} = 1$ 이 된다. 마찬가지로 C_w $h = i^1, \dots, i^p$ 가 복합노드 C_w 의 모든 전제노드의 집합이면 [정의 1] 및 [정의 2]에 의해 $R_{hw} = 1, h = i^1, \dots, i^p$ 이고 $S_w = p$ 가 된다. 여기서 $T_h^k = 1, h = i^1, \dots, i^p$ 이면 식 (2)에 의해

$$U_w^k = \sum_{r=1}^n T_r^k \cdot R_{rw} \geq \sum_{h=i^1}^p T_h^k \cdot R_{hw} (= p) \geq S_w (= p)$$

가 성립하여 식 (3)에 의해 $T_w^{k+1} = 1$ 이 된다 ■

[명제 2] 초기 진리상태 벡터를 T^0 , 이를 바탕으로 식 (2) 및 식 (3)의 연속적인 적용을 통해 얻어진 최종 진리상태벡터를 T^F 라 하고 S_j 및 S_k 를 각각 [정의 2]에 정의된 충분조건벡터 S의 j번째 및 k번째 요소를 나타낸다고 하자. 그러면 임의의 진리상태 벡터 T가 T^F 가 되기 위한 필요충분조건은 다음과 같다 :

- (1) 모든 리프노드 C_i 에 대해 $T_i \equiv T_i^0$ 이다. 즉 T에 나타난 리프노드의 진리상태는 T^0 와 일치한다.
- (2) T에 참이라고 나타난 모든 중간 또는 루트노드 C_j 에 대해 다음 사항이 성립한다:
(C_j 의 전제노드 중 T에 참이라고 나타난 것의 개수) $\geq S_j$.
- (3) T에 거짓이라고 나타난 모든 중간 또는 루트노드 C_k 에 대해 다음 사항이 성립한다:
(C_k 의 전제노드 중 T에 참이라고 나타난 것의 개수) $< S_k$ ■

본 논문에서 제시하고자 하는 IMITAO는 [명제 2]의 (조건 1) 및 (조건 3)을 만족하는 초기 LAS 진리상태벡터 T^0 를 구한 후 (조건 2)를

충족하지 않는 노드들의 진위값을 단계적으로 바꾸어 줌으로써 최종 진리상태벡터 T^F 를 구하는 방식이다. 따라서 IMITAO 추론과정은 크게 추론을 위한 초기진리상태벡터 $\overline{T^0}$ 를 구하는 단계와 이를 바탕으로 최종 진리상태벡터 T^F 를 구하는 단계로 나뉘어진다.

• 단계 1 : 초기해 $\overline{T^0}$ 를 계산하는 단계

IMITAO를 위한 초기 진리상태벡터, 즉 [명제 2]의 (조건 1)과 (조건 3)을 만족하는 초기해 $\overline{T^0}$ 를 구하는 방법은 여러가지가 있을 수 있다. 가장 간단하게는 초기 진리상태벡터 T^0 에서 중간노드 및 루트노드의 진위값을 모두 1로 바꾸어 주면 된다. 그런데 초기해 $\overline{T^0}$ 는 최종 진리상태벡터 T^F 에 가까울수록 다시말해 [명제 2]의 (조건 1) 및 (조건 3)을 만족하면서 (조건 2)를 만족하지 않는 요소가 적으면 적을수록 효율성이 높다고 할 수 있다. 이런 관점에서 볼 때 앞서 언급한 간단한 방식에 의해 구해진 $\overline{T^0}$ 는 매우 비효율적이다. 본 논문에서 제시하는 IMITAO의 효율성은 초기해 $\overline{T^0}$ 의 효율성에 크게 좌우된다. 따라서 본 논문에서는 주어진 TAOG에 나타난 규칙경로(modus ponens chain or implication path)를 이용하여 효율적인 초기해 $\overline{T^0}$ 를 구하는 방식을 제시하고자 한다. 우선 1차 경로행렬 P를 다음과 같이 정의하자.

$$P_{ij} = \begin{cases} 1, & \text{if } C_i \rightarrow C_j \text{ and } i \neq j \\ 0, & \text{otherwise} \end{cases},$$

$$i, j = 1, \dots, n.$$

이와 같이 정의된 1차 경로행렬 P의 각 요소 P_{ij} 는 주어진 TAOG에서 노드 C_i 로부터 노드 C_j 로 연결되는 호가 존재하는가의 여부를 나타낸다. 즉 호가 존재하면 1이고 없으면 0이다. 즉 행렬 P는 [정의 1]에 정의된 규칙행렬 R의 대각선 요소를 모두 0으로 바꾼 결과와 일치한

다(<그림 4-a> 참조). 이제 1차 경로행렬 P의 곱 P^k 의 의미에 대해 살펴보자. 우선 행렬 P의 제곱인 2차 경로행렬 P^2 는 어떤 의미를 가지는가? P_{ij}^2 를 행렬 P^2 의 i번째 행과 j번째 열에 해당하는 요소라고 하면 이는 노드 C_i 로부터 노드 C_j 로 2단계만에 연결되는 경로의 수를 의미한다. 참고로 <그림 1>의 예제 TAOG에 대한 P^2 이 <그림 4-b>에 나타나 있다. <그림 4-b>에서 P_{67}^2 이 2 라는 것은 노드 C_6 로부터 노드 C_7 로 2단계만에 연결되는 경로가 두 개 ($C_6 \rightarrow C_9 \rightarrow C_7$ 및 $C_6 \rightarrow C_{10} \rightarrow C_7$)라는 것을 의미한다. 이러한 논의를 일반화하면 k차 경로행렬 P^k 는 노드간에 k개의 호를 통해 연결되는 경로의 수를 표시한다는 것을 알 수 있다. 또한 L을 주어진 TAOG에 존재하는 최대경로라고 하면(<그림 1>의 예제 TAOG에 대한 L은 4가 됨), P^{L+1} 은 모든 성분이 0인 행렬이 됨을 유추할 수 있다.

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

(a) 1차 경로행렬

$$A^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

(b) 2차 경로행렬

<그림 4> <그림 1>의 예제 TAOG에 대한 1, 2차 경로행렬

이제 총 경로행렬 Q를 다음과 같이 정의 하자 :

$$Q = \sum_{l=0}^L P^l,$$

여기서 L은 최대경로의 수, P^0 는 I를 의미함 ■

이렇게 정의된 총경로행렬 Q의 모든 대각선 요소는 1이 되며, 대각선 요소가 아닌 Q_{ij} ($i \neq j$)는 노드 C_i 로부터 노드 C_j 로 연결되는 모든 경로의 수를 의미한다. <그림 1>에 나타난 예제 TAOG에 대한 총경로행렬 Q는 <그림 5>에 예시되어 있다.

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 2 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 2 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 2 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

<그림 5> <그림 1>의 예제 TAOG에 대한 총경로 행렬 $Q = I + P + P^2 + P^3 + P^4$

효율적인 초기해 T^0 를 구하기 위해서는 이 상에서 정의된 총경로행렬 Q 외에 다음과 같이 최소경로벡터에 대한 정의가 필요하다.

[정의 4] 최소경로벡터 M : $C_i, i=1, \dots, n$ 을 주어진 TAOG에 나타난 노드들이라 하자. 이 경우 $1 \times n$ 최소경로벡터 $M = [M_1, \dots, M_n]$ 은 다음과 같이 정의된다. 우선 리프노드 C_i 에 대한 최소경로 M_i 는 항상 1로 정의된다. 노드 C_i 가 중간 또는 루트노드일 경우 M_i 는 노드 C_i 가 추론을 통해 참이되기 위하여 반드시 참이 되어야 하는 리프노드의 집합으로부터 노드 C_i 에 이르는 경로(path)의 수중 최소값을 나타

낸다. 예를들어 <그림 1>에서 노드 C_6 이 참이 되기 위해 반드시 참이 되어야 하는 리프노드의 집합은 $\{C_2, C_3\}$ 또는 $\{C_4\}$ 이다. 그리고 집합 $\{C_2, C_3\}$ 에 속하는 노드로부터 노드 C_6 에 이르는 경로는 두 개($C_2 \rightarrow C_8 \rightarrow C_6$ 및 $C_3 \rightarrow C_8 \rightarrow C_6$)이고, 집합 $\{C_4\}$ 에 속하는 노드로부터 노드 C_6 에 이르는 경로는 $C_4 \rightarrow C_6$ 하나이다. 따라서 노드 C_6 에 대한 최소경로 M_6 의 값은 1이 된다. 이러한 의미를 나타내는 최소경로벡터 M을 보다 엄밀하게 정의하면 다음과 같다 :

(1) C_i 가 리프노드인 경우 : $M_i = 1,$

(2) C_i 가 리프노드가 아닌 경우 :

$C_j, j = i^1, \dots, i^j$ 를 노드 C_i 에 대한 전제 노드들의 집합이라 하자. 그러면,

(i) C_i 가 단순노드인 경우 :

$$M_i = \text{MIN}_{j=i^1}^{i^j} \{M_j\}$$

(ii) C_i 가 복합노드인 경우 :

$$M_i = \sum_{j=i^1}^{i^j} M_j \quad \blacksquare$$

따라서 n개의 노드 $C_i, i=1, \dots, n$ 로 구성된 TAOG가 주어진 경우 $M_i, i=1, \dots, n$ 는 위의 정의를 이용하여 TAOG의 하위 레벨에 속하는 노드부터 시작하여 차례로 구해 나가면 된다. 예를 들어 <그림 1>의 예제 TAOG에 대한 최소경로벡터 M은 다음과 같이 구할 수 있다 :

$$\text{리프노드} : M_1 = M_2 = M_3 = M_4 = M_5 = 1$$

$$M_8(\text{복합노드}) = M_2 + M_3 = 2$$

$$M_6(\text{단순노드}) = \text{Min} \{M_4, M_8\} = 1$$

$$M_9(\text{복합노드}) = M_1 + M_6 = 2$$

$$M_{10}(\text{복합노드}) = M_6 + M_5 = 2$$

$$M_7(\text{단순노드}) = \text{Min} \{M_9, M_{10}\} = 2 \Rightarrow$$

$$M = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 2 \ 2 \ 2 \ 2].$$

이상에서 설명한 총경로행렬 Q 및 최소경로벡터 M이 주어졌을 경우 이를 바탕으로 초기해 \overline{T}^0 를 구하는 절차는 다음과 같다. 추론을 위해 실험 또는 관측으로부터 얻어진 초기 진리상태를 T^0 라 하자. 그리고 벡터 W를 T^0 와 총경로행렬 Q의 곱, 즉

$$W = T^0 \cdot Q \quad (4)$$

라 하자. 그러면 W의 각 요소 $W_i, i=1, \dots, n$ 는 C_i 가 리프노드일 경우에는 $W_i = T_i^0$ 가 되고, C_i 가 중간 또는 루트노드일 경우에는 T^0 에 참이라고 나타난 모든 리프노드로부터 노드 C_i 로 연결되는 모든 경로의 수를 의미한다. 이제 벡터 \overline{T}^0 를 다음과 같이 정의하면,

$$\overline{T}^0 = \begin{cases} 1, & \text{if } W_i \geq M_i \\ 0, & \text{otherwise} \end{cases}, i = 1, \dots, n, \quad (5)$$

\overline{T}^0 는 다음 명제에서 알 수 있듯이 IMITAO를 위한 초기 진리상태벡터가 되기 위한 조건, 즉 [명제 2]의 (조건 1) 및 (조건 3)을 만족한다.

[명제 3] (식 4) 및 (식 5)에 의해 구해진 \overline{T}^0 는 [명제 2]의 (조건 1) 및 (조건 3)을 만족한다.

[증명] 우선 (조건 1) 즉 임의의 리프노드 C_i 에 대해 $\overline{T}_i^0 = T_i^0$ 가 성립함을 증명해 보자. 식 (4)에 의해 $W_i = \sum_{h=1}^n T_i^0 \cdot Q_{hi}$ 이다. 그런데 C_i 가 리프노드이므로 $h \neq i$ 일 경우 $Q_{hi} = 0$ 이다. 따라서 $W_i = T_i^0 \cdot Q_{ii} = T_i^0 \cdot 1 = T_i^0$ 이 성립한다. 또한 리프노드 C_i 에 대한 $M_i = 1$ 이므로 식 (5)에 의해

$$[T_i^0 = 1] \Rightarrow [W_i = 1] \Rightarrow [W_i \geq M_i (=1)] \Rightarrow$$

$$[\overline{T}_i^0 = 1] \text{ 및}$$

$$[T_i^0 = 0] \Rightarrow [W_i = 0] \Rightarrow [W_i < M_i (=1)] \\ \Rightarrow [\overline{T}_i^0 = 0]$$

이 되어 항상 $\overline{T}_i^0 = T_i^0$ 가 성립함을 알 수 있다.

이제 (조건 3)이 성립함을 증명해 보자. 노드 C_j 를 임의의 중간 또는 루트노드라 하고 CNT_j 를 C_j 의 전제노드 중 \overline{T}^0 에 참이라고 나타난 것의 개수라 할 때 (조건 3)의 내용은 $[\overline{T}_j^0 = 0] \Rightarrow [CNT_j < S_j]$ 가 성립한다는 것이다. 이는 $[CNT_j \geq S_j] \Rightarrow [\overline{T}_j^0 \neq 0]$ 와 동일하고 $[\overline{T}_j^0 \neq 0] \equiv [W_j \geq M_j]$ 이므로 (조건 3)을 증명하기 위해서는 $[CNT_j \geq S_j] \Rightarrow [W_j \geq M_j]$ 가 성립함을 증명하면 된다.

임의의 리프노드로부터 노드 C_j 에 이르는 경로는 반드시 C_j 의 전제노드를 거치게 된다. 따라서 C_j 의 전제노드의 집합을 $C_{h^i}, h = i^1, \dots, i^j$ 라 하면 $W_j = \sum_{h=i^1}^{i^j} W_h$ 가 된다. 이를 바탕으로 C_j 가 단순노드일 경우와 복합노드일 경우로 나누어 $[CNT_j \geq S_j]$ 가 성립한다는 가정하에 $[W_j \geq M_j]$ 가 성립함을 증명하면 다음과 같다.

(i) C_j 가 단순노드일 경우

C_j 가 단순노드일 경우 $S_j = 1$ ([정의 2]), $M_i = \text{Min}\{M_{i^1}, \dots, M_{i^j}\}$ ([정의 4])이 된다. 여기서 $CNT_j \geq S_j$ 가 성립한다는 것은 $W_{h^i}, h = i^1, \dots, i^j$ 중 $M_{h^i}, h = i^1, \dots, i^j$ 보다 같거나 큰 것이 최소한 하나 이상 존재한다는 것을 의미한다. 따라서 이 경우

$$W_j = \sum_{h=i^1}^{i^j} W_h \geq \text{Min}\{M_{i^1}, \dots, M_{i^j}\} = M_i$$

가 성립한다.

(ii) C_j 가 복합노드일 경우

C_j 가 복합노드일 경우 S_j 는 C_j 의 모든 전제노드의 수가 되고([정의 2]), $M_i = \sum_{h=i^1}^{i^j} M_h$ ([정의 4])가 된다. 따라서 $CNT_j \geq S_j$ 가 성립한다는 것은 모든 $W_h, h = i^1, \dots, i^j$ 가 $M_h, h = i^1, \dots, i^j$ 보다 같거나 크다는 것을 의미한다. 따라서 이 경우

$$W_j = \sum_{h=i^1}^{i^j} W_h \geq \sum_{h=i^1}^{i^j} M_h = M_j \text{가 성립한다} \blacksquare$$

앞에서 사용한 예(<그림 1>의 예제 TAOG에서 초기 진리상태벡터 $T^0 = [1111000000]$ 일 경우)에 대한 W 및 $\overline{T^0}$ 를 예시하면 다음과 같다:

(식 4) $W = T^0 \cdot Q = [11110025232]$

(식 5) (W 를 M 과 비교하여) $\overline{T^0} = [11110011111]$.

• 단계 2 : 초기해 $\overline{T^0}$ 를 바탕으로 최종진리상태 T^F 를 구하는 단계

단계 1에서 구한 초기해 $\overline{T^0}$ 를 기초로 하여, 최종 진리상태벡터 T^F 를 구하는 절차를 제시하기 위해서는 다음과 같이 IMITAO용 규칙행렬 \overline{R} 에 대한 정의가 필요하다.

[정의 5] IMITAO용 규칙행렬 $\overline{R}: C_i, i=1, \dots, n$ 을 주어진 TAOG에 나타난 노드들이라 하고, C_i 로부터 C_j 로 가는 호가 있을 경우 이를 $C_i \rightarrow C_j$ 로 나타낸다고 하자. 그러면 n 개의 노드로 이루어진 TAOG에 대한 $n \times n$ IMITAO용 규칙행렬 \overline{R} 는 다음과 같이 정의된다 :laslbs

$$R_{ij}(i \neq j) = \begin{cases} 1, & \text{if } C_i \rightarrow C_j \\ 0, & \text{otherwise} \end{cases}, \quad i, j=1, \dots, n,$$

$$R_{ii} = \begin{cases} 1, & \text{if } C_i \text{가 리프노드인 경우} \\ 0, & \text{otherwise} \end{cases} \\ i=1, \dots, n \blacksquare$$

이상과 같이 정의된 IMITAO용 규칙행렬 \overline{R} 는 [정의 1]에 나타난 규칙행렬 R 의 대각선 요소중 중간 또는 루트노드에 대한 값을 0으로 바꾸어 준 결과와 일치한다. 예를들어 <그림 1>의 예제 TAOG에 대한 IMITAO용 규칙행렬 \overline{R} 가 <그림 6>에 나타나 있다.

$$\overline{R} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

<그림 6> <그림 1>의 예제 TAOG에 대한 \overline{R}

이제 이상에서 정의된 IMITAO용 규칙행렬 \overline{R} , 진리상태벡터 \overline{T} 및 [정의 2]에 정의된 충분조건벡터 S 를 이용한 역방향 추론 절차를 설명하면 다음과 같다. $\overline{T^k} = (\overline{T_1^k}, \dots, \overline{T_n^k})$ 를 IMITAO에 의한 현재의 진리상태라 하고, $\overline{U^k}$ 를 벡터 $\overline{T^k}$ 와 \overline{R} 의 곱 즉,

$$\overline{U^k} = \overline{T^k} \cdot \overline{R} \tag{6}$$

라 하자. 그러면 $\overline{U^k}$ 의 각 성분 $\overline{U_i^k}$ 는 노드 C_i 의 전제노드 중 $\overline{T^k}$ 에 참이라고 나타난 것의 수를 의미하게 된다. 그리고 $\overline{T^{k+1}} = (\overline{T_1^{k+1}}, \dots, \overline{T_n^{k+1}})$ 을

$$\overline{T_i^{k+1}} = \begin{cases} 1, & \text{if } \overline{U_i^k} \geq S_i \\ 0, & \text{otherwise} \end{cases}, \quad i=1, \dots, n \tag{7}$$

이라고 정의하면, 진리상태 \overline{T}^k 에 값이 1으로 나타난 노드 중 [명제 2]의 (조건 2)를 충족하지 않는 노드는 진리상태 \overline{T}^{k+1} 에서 값이 0으로 나타나게 된다. 따라서 IMITAO를 통해 최종 진리상태벡터 T^F 를 구하기 위해서는 초기해 \overline{T}^0 을 바탕으로 식 (6) 및 식 (7)을 이용하여 \overline{T}^k , $k=1, 2, \dots$ 를 진리상태가 더 이상 변하지 않을 때까지 계속 구해 나가면 된다. 단계 1에서 구한 초기해 $\overline{T}^0 = [1110011111]$ 을 바탕으로 이상에서 제시한 IMITAO의 모든 추론 과정을 예시하면 다음과 같다:

식 (6) $\overline{U}^0 = \overline{T}^0 \cdot \overline{R} = [1110012221]$

식 (7) $\overline{T}^1 = [1110011110]$

식 (6) $\overline{U}^1 = \overline{T}^1 \cdot \overline{R} = [1110011221]$

식 (7) $\overline{T}^2 = [1110011110] = \overline{T}^1$

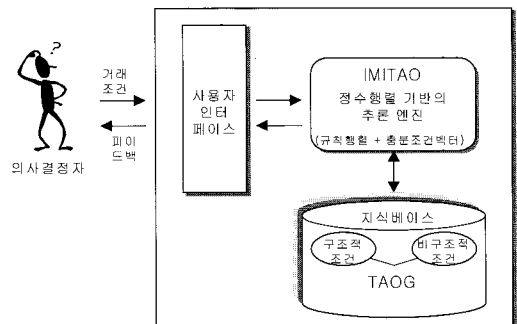
따라서 \overline{T}^1 이 IMITAO에 의해 구하고자 하는 최종 진리상태벡터가 된다).

IV. 실험 및 의의: 기업간 전자상거래 응용사례

본 장에서는 지금까지 서술한 TAOG 및 IMITAO를 활용하여 전자상거래에서 구매기업과 공급기업간 계약체결을 위한 의사결정과정을 체계적으로 분석하여 보자. 우선, 기업간 전자상거래 문제에서는 여러 가지 고려하여야 할 요소가 많이 있다. 즉, 첫째로는 두 거래 당사자가 동시에 협상을 진행하고자 하는 객관적인(또는 구조적인) 요인들이 있다. 예를들어 가격조건, 수량, 지불조건, 배달 등이 여기에 해당된다. 둘째로는 이같은 객관적인 요인에 영향을 미치는

주관적인 (또는 비구조적인) 요인들이 있다. 예를들어 원자재 수급여건, 노조와의 관계 등이 여기에 해당된다. 이는 당사자에만 해당되는 요인들이지만 해당 기업의 객관적인 요인에 궁극적으로 영향을 미친다는 점에서 기업간 전자상거래에 관한 의사결정지원에서 반드시 고려되어야 한다.

따라서, 기업간 전자상거래에 관한 의사결정 지원을 위해서는 이같은 객관적인 요인과 주관적인 요인을 모두 동시에 고려하면서도 아울러 신속한 추론이 될 수 있는 시뮬레이션 메커니즘이 필요하다. 이러한 요구조건을 고려할 때에 3장에서 제안한 IMITAO는 이상의 요구조건을 모두 충족시킬 수가 있다. <그림 7>은 본 연구에서 제안하는 TAOG와 IMITAO를 적용한 기업간 전자상거래 추론과정을 도시한 것이다. 즉, 거래기업들은 자기가 생각하는 거래조건과 주관적 판단등을 TAOG로 표현하고, 이를 IMITAO를 통하여 정수행렬에 의한 추론으로 상대방 기업과 인터넷을 통하여 다양한 시뮬레이션을 할 수가 있다. 특히 해당 기업의 지식베이스는 지식관리자가 생성할 수도 있고, 또는 TAOG를 바탕으로 자동으로 생성될 수도 있기 때문에 추론할 때마다 일일이 새로 작성할 필요는 없다.



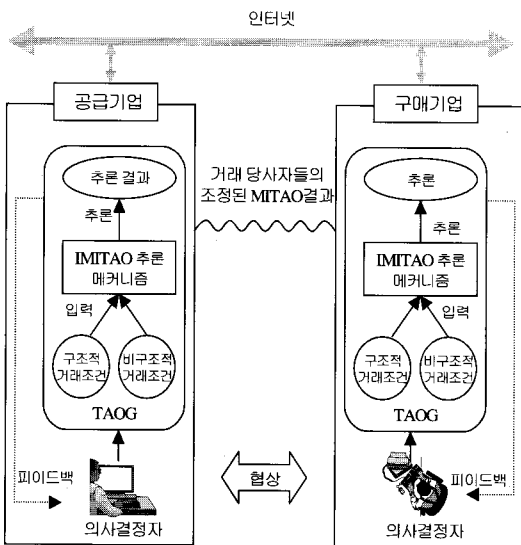
<그림 7> TAOG와 IMITAO를 이용한 추론방법의 구조

한편, 이러한 지식베이스를 기초로 하여 상대방 기업이 제시하는 구조적 조건을 받아들일지

1) 지금까지 언급한 IMITAO는 Lee & Cho(1999)에서 언급한 추론 메커니즘의 효율성을 보다 확장한 개념이다.

여부를 결정하고자 할 때에 해당 기업이 가지고 있는 다양한 비구조적 조건들을 체계적으로 반영할 수가 있다. 이러한 과정을 반복하면 각 기업의 입장에서 받아들일 수 있는 합리적인 거래조건을 찾아낼 수가 있는 장점이 있다. 특히 본 연구에서 제안하는 IMITAO를 이용하여 추론을 하면 인터넷 공간에서도 얼마든지 실시간으로 신속한 추론이 가능하며, 인터넷에서 이른바 B-to-B(기업대 기업간) 전자상거래를 효과적으로 추진할 수 있는 실제적인 추론엔진을 설계할 수가 있다.

이상의 내용을 정리하여 기업간 전자상거래 문제를 해결하는 과정을 그림으로 도시하면 <그림 8>과 같다. 여기에서는 기업간 전자상거래 기업들이 서로의 계약 조건을 입력하고 IMITAO를 이용하여 의사결정에 도달하는 과정을 흐름별로 제시하고 있다.



<그림 8> TAOG와 IMITAO에 의한 기업간 전자상거래 문제해결과정

4.1 문제제시 및 분석

본 절에서 소개하는 사례에서는 기업간 전자상거래를 하고자 하는 두 기업이(하나는 공급기

업이라 하고 또 하나는 구매기업이라고 가정한다) IMITAO를 활용한 시뮬레이션을 통하여 협상을 하므로써 최적의 거래조건을 찾아내어 계약체결에까지 이르는 과정을 보여주고 있다. 본 사례에서 소개하는 기업간 전자상거래 문제는 KINET(주)에서 사용된 실제문제를 기초로 수정한 것이다.

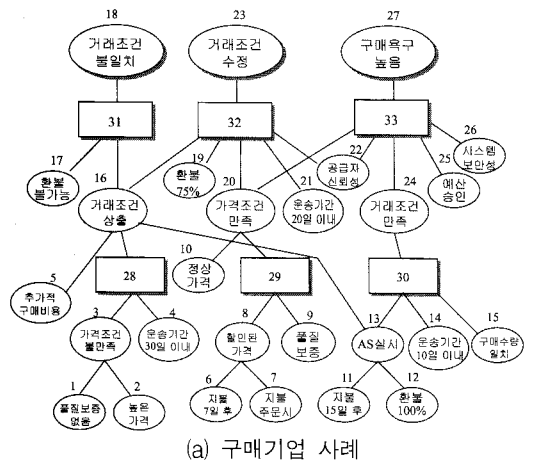
본 사례에서 두 기업은 각각 객관적(즉, 구조적) 거래조건과 비구조적(즉, 비구조적) 거래조건을 가지고 협상에 임한다. 이미 설명한 바대로 기업간 전자상거래에서 실제 협상의 대상이 되는 것은 구조적 거래조건이지만, 이러한 구조적 거래조건에 영향을 주는 것은 비구조적 거래조건이다. 이를 보다 이해하기 쉽게 설명하면 구조적 거래조건은 계량화하여 표현할 수 있는 거래조건으로서 기업간 전자상거래시 서로 협상의 대상으로 하는 일반적인 조건들을 의미한다. 예를들어 가격, 수량, 지불조건, 납기일 등이 대표적인 구조적 거래조건이라고 할 수 있다. 반면에 비구조적 거래조건이란 의사결정자인 해당 기업만이 갖는 특수한 정성적 조건들을 의미한다. 따라서, 이는 계량화 할 수 없는 정성적인 거래조건들이지만, 해당기업에서 상대방 기업이 제시하는 구조적 거래조건을 받아들이나 여부를 결정할 때에 반드시 고려하여야 할 조건들을 의미한다. 가장 대표적인 비구조적 조건중의 하나가 노동조합의 요구사항, 원자재 수급조건, 예산 승인 등이 있다.

기업간 전자상거래에서 고려하여야 할 거래조건을 조금 더 세부적으로 살펴보면 다음과 같다. 먼저 공급기업은 계약의 체결을 위하여 구매기업에서 제시한 가격, 운송기간, 공급수량, 구매기업신용정도, 운송기간 및 운송수단확보, 원재료 및 노동력확보, 노동조합승인 등의 조건을 통합적으로 고려하여 구매기업과 구조적인 거래조건들을 협상할 수 있다. 구매기업 역시 공급기업에서 제시한 수량, 가격 및 할인정도, 지불조건, 운송기간, 품질보증여부, 환불기간, 공

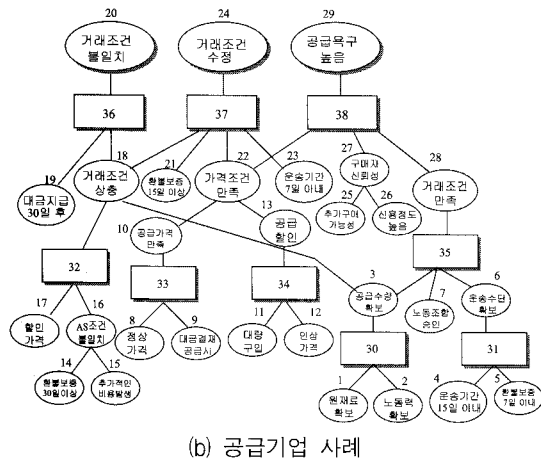
급기업의 신뢰성, 추가적인 구매비용, 예산집행 등의 조건을 고려하여 공급기업과 구조적 거래 조건들을 협상하여야 한다.

4.2 지식베이스의 구축

문제의 분석이 끝났으면 앞에서 제시한 거래 조건을 <그림 9>에서와 같이 TAOG 형태로 표현하여야 한다. 그리고 이러한 각 기업의 TAOG를 기초로 하여 IMITAO에 필요한 규칙행렬(<그림 10>)과 총경로행렬(<그림 11>) 등을 계산한다. 물론 이같은 행렬들이 IMITAO에서 필요한 지식베이스의 역할을 한다.



(a) 구매기업 사례



(b) 공급기업 사례

<그림 9> 기업간 전자상거래의 사례를 위한 TAOG

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----

(a) 구매기업의 규칙행렬

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----

(b) 공급기업의 규칙행렬

<그림 10> 규칙행렬

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----

(a) 구매기업의 총경로행렬

(This area contains a large grid of small, illegible text or symbols, likely a data matrix or a very small table.)

이때 <그림 10, 11>에 나타나 있는 각 구매 기업의 노드값($C_i, i = 1, 2 \dots 33$)과 공급기업의 노드값($C_i, i = 1, 2 \dots 38$)은 기업간 전자상거래에서 거래당사자의 거래조건을 의미한다. 보다 구체적으로 이 값이 의미하는 바는 <표 1>에 요약이 되어있다.

이상과 같이 규칙행렬과 총경로행렬을 구하였으면, 효율적인 초기해 계산을 위하여 최소경로 벡터를 구하여야 한다. 이를 3장에서 소개한 내용을 기초로 계산하면 다음과 같다.

(b) 공급기업의 총경로행렬
<그림 11> 총경로행렬

• 구매기업 최소행렬벡터($M_{\text{구매기업}}$)=
[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1

<표 1> 노드값이 의미하는 내용

구매기업의 거래 제약조건		공급기업의 거래 제약조건	
노드	설명	노드	설명
C1, C9	구매하는 물건에 대한 국제규격의 품질보증 획득 여부	C1, C2, C3	공급수량을 확보하기 위한 원재료 및 노동력의 확보정도
C2, C3, C8, C10	공급기업이 제시한 가격조건	C4, C6, C23	구매기업이 제시하는 운송기간 조건 및 이를 충족시키기 위한 운송수단확보 여부
C5	추가적인 구매비용	C8, C12, C17	구매기업이 원하는 물품가격
C6, C7, C11	구매시 공급기업에서 제시한 대금지불조건	C11, C13	구매기업의 대량구입 및 공급기업의 가격할인 여부
C12, C17, C19	공급기업이 제시하는 환불여부 및 기간	C9, C10, C19	구매기업이 원하는 대금지불 조건
C4, C14, C21	주문한 물건의 운송기간 조건	C5, C14, C21	구매기업이 물건을 구입하였을 때 원하는 환불여부 및 보증기간 정도
C22	공급기업에 대한 신뢰정도	C25, C26, C27	구매기업의 추가적인 구매 및 신용정도
C25	구매액에 대한 예산집행여부	C15	관세나 추가적인 운송비용 부담 정도
C26	거래시 구매기업 정보에 대한 공급기업의 보안시스템 제공여부	C7	노동조합의 승인
C16	거래조건에 대한 상충관계 여부	C18	구매기업에서 제시한 거래조건에 대한 상충관계 여부
C20	공급기업이 제시하는 가격조건 및 지불조건에 따른 만족여부	C22	구매기업에서 제시하는 가격조건 및 할인조건에 대한 만족여부
C24	공급기업 제시하는 수량, 운송기간 및 환불조건에 따른 거래조건 만족여부	C28	거래조건 만족여부
C28 ~ C33	복합노드	C30 ~ C38	복합노드
C18	공급기업과의 거래조건이 일치하지 않음	C20	구매기업과의 거래조건이 일치하지 않음
C23	운송기간, 환불조건, 지불조건 등에 대한 거래조건 수정을 요구	C24	운송기간, 환불, 지불 등에 대한 거래조건 수정을 요구
C27	구매욕구가 높다는 뜻으로 계약이 체결될 가능성이 높음	C29	공급욕구가 높다는 뜻으로 계약이 체결될 가능성이 높음

1 1 1 5 3 1 1 7 2 2 3 2 5 7

- 공급기업 최소행렬벡터($M_{\text{공급기업}}$)=
 [1 1 2 1 1 2 1 1 1 1 2 1 1 1 1 2 1
 3 1 2 1 6 1 1 1 5 9 2 2 2 2 2 5 3 6 9]

4.3 추론과정

4.2에서 규칙행렬과 총경로행렬로 구축된 지식베이스를 활용하여 기업간 전자상거래 당사자들이 적정한 구조적 거래조건을 추론하는 과정을 살펴보자. 우선 공급기업이 구매기업에 제시한 구조적 거래조건이 다음과 같다고 가정하자.

첫번째 거래조건 <공급기업→구매기업>	
• 가격 :	표시된 정찰가격
• 구매수량 :	구매기업이 원하는 수량 범위내
• 운송기간 :	주문 후 30일 이내 운반
• 할인 :	해당사항 없음
• 대금지불 :	주문시 즉시 지불
• 품질 :	품질인증 없음
• 기타사항 :	거래에 따른 추가적인 비용발생
• 환불 :	환불 및 반품은 불가능

이같은 공급기업의 구조적 거래조건을 기초로 하여 구매기업은 [그림 9]의 (a)에 나타나 있는 TAOG와 이를 기초로 한 규칙행렬 [그림 10-a] 및 총경로행렬 [그림 11-a]을 가지고 다음과 같이 추론 시뮬레이션을 할 수가 있다.

C15	구매수량은 일치합니까?	Yes	1
C4	운송기간은 주문 후 10일 이내입니까?	No	0
C10	대량구입시 가격할인이 가능합니까?	No	0
C5	추가적인 구매비용이 발생합니까?	Yes	1
C17	구매한 물품은 환불 및 반품이 가능합니까?	No	0
C7	구매대금은 주문 즉시 지불합니까?	Yes	1
C1	구매하는 물품은 품질인증을 획득했습니까?	No	0

위의 추론과정을 바탕으로 추론메카니즘의 초기값으로 각 거래조건에 해당되는 입력노드에 0과 1을 대입한다. 그러면 이를 바탕으로 구매기업의 지식베이스를 통하여 추론이 이루어지는데, 그 추론과정이 <그림 12>에 제시되어 있다. <그림 12>에서 C18, C23, C27은 각각 순서대로 거래조건 불일치, 거래조건수정, 구매욕구 높음을 의미한다. 따라서, 본 예제의 경우 최종 진리상태인 T^9 에서 C18이 1이므로 공급기업과의 거래조건이 일치하지 않는다는 결론을 얻었다. 결국 구매기업은 수정된 구조적 거래조건을 공급기업에게 제시하여야 한다.

구매기업은 우선 "거래조건 불일치"라는 이유가 나오게 된 이유를 분석한 뒤 구매가격, 지불조건, 반품 및 환불, 운송기간 등의 거래조건에 대하여 구매기업에서 원하는 수준으로 다음과 같이 거래조건을 수정하여 공급기업에게 제시한다.

공급기업에서는 구매기업이 제시한 새로운 수

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22	C23	C24	C25	C26	C27	C28	C29	C30	C31	C32	C33
T0	1	0	0	1	1	0	1	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	5	3	1	1	7	2	2	3	2	5	7
W	1	0	1	1	1	0	1	1	0	1	0	0	0	0	1	3	1	4	0	2	0	0	5	1	0	0	3	2	1	1	4	5	3
T ⁰	1	0	1	1	1	0	1	1	0	1	0	0	0	0	1	1	1	1	0	1	0	0	1	0	0	0	1	0	0	1	1	0	
U ⁰	1	0	1	1	1	0	1	1	0	1	0	0	0	0	1	2	1	1	0	1	0	0	1	0	0	0	0	2	1	1	2	2	1
S	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	3	3	2	5	5
T ¹	1	0	1	1	1	0	1	1	0	1	0	0	0	0	1	1	1	1	0	1	0	0	1	0	0	0	1	0	0	1	0	0	
U ¹	1	0	1	1	1	0	1	1	0	1	0	0	0	0	1	2	1	1	0	1	0	0	0	0	0	0	0	2	1	1	2	2	1
T ²	1	0	1	1	1	0	1	1	0	1	0	0	0	0	1	1	1	1	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0

<그림 12> 구매기업의 IMITAO 추론결과 - "거래조건 불일치"

정거래조건에 대하여 IMITAO를 적용하여 추론을 한후에 과연 이같은 수정거래조건을 받아들여야 하는지 여부를 결정한다.

두 번째 거래조건 <구매기업 →공급기업>	
•가	격: 표시된 정찰가격
•	구매수량: 대량구입가능하며 이때 가격할인을 원함
•	운송기간: 주문 후 5일 이내
•	대금지불: 주문 후 10일 이내 지급가능
•	기타사항: 관세나 추가운송비용을 부담가능
•	품 질: 품질인증 보장
•	환불 및 반품정도: 환불 및 보증은 물건 수령 후 20일까지

- C8 공급가격은 표시된 정찰가격입니까? Yes 1
- C23 운송기간은 주문 후 7일 이내입니까? Yes 1
- C21 환불 및 반품의 기간은 15일 이상입니까? Yes 1
- C17 대량공급시 가격할인이 있습니까? Yes 1
- C9 구매대금은 주문 즉시 지불합니까? Yes 1
- C26 구매기업의 신용도가 높습니까? No 0
- C15 추가적인 운송비용이 발생합니까? Yes 1
- C1 구매기업에서 주문한 주문량을 확보하기 위 한 원재료확보가 가능합니까? Yes 1
- C2 납기일을 맞출 수 있는 노동력의 확보가 가 능합니까? Yes 1

그러면 이를 바탕으로 공급기업의 추론메커니즘에 초기값을 입력하면 추론과정은 <그림 13>과 같이 실시간으로 이루어진다.

<그림 13>에서 루트노드인 C20, C24, C29은 각각 순서대로 거래조건 불일치, 거래조건수정, 공급욕구 높음을 의미한다. 본 추론의 결과 최종 진리상태인 T^2 의 결과값인 C24이 1이 제시되었다. 앞의 <표 1>에서 각 노드값에 대하여 설명할 때 공급기업의 루트노드 중의 하나인 C24노드는 운송기간, 지불조건, 환불 등의 거래조건을 수정하면 공급욕구가 향상될 수 있음을 의미한다고 설명하였다. 이에 공급기업에서는 협상중인 구매기업과 거래를 성사시키기 위하여 구매기업의 거래조건을 분석하고 그 결과를 토대로 하여 다시 한번 구매기업과 협상을 하기로 하였다. 거래조건 분석결과 구매기업에서 제시한 조건 중에서 운송기간 조건에서 서로 상충관계가 발생하였는데 그 이유는 구매기업에서 대량구입을 원함으로 주문 후 5일 이내에 구매기업에서 주문물량을 공급기업에서 생산할 수 있는 노동력의 확보가 어렵고, 이에 따른 노동조합과의 마찰이 예상되었다. 또한 구매기업의 환불기간 조건 또한 구매기업측에서 요구하는 20일은 너무 무리한 조건으로 분석되어 공급기업에서 가능한 조건인 물건수령 후 7일로 변경하였다. 공급기업에서는 이상의 수정된 조건을 구매기업에 제시한다.

그러면 구매기업은 공급기업에서 제시한 거래조건을 바탕으로 본 다음과 같은 입력에 의하여 추론이 이루어진다.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22	C23	C24	C25	C26	C27	C28	C29	C30	C31	C32	C33	C34	C35	C36	C37	C38
T0	1	1	0	0	0	0	1	1	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M	1	1	2	1	2	1	1	1	2	1	1	1	2	1	1	1	1	2	1	3	1	2	1	6	1	1	1	5	9	2	2	2	2	2	5	3	6	9
W	1	1	2	0	0	0	1	1	2	0	0	0	0	0	1	1	1	4	0	4	1	2	1	8	0	0	0	2	4	2	0	2	2	0	2	4	8	4
T ⁰	1	1	1	0	0	0	1	1	1	0	0	0	0	0	1	1	1	1	0	1	1	1	1	1	0	0	0	0	0	1	0	1	1	0	0	1	1	0
U ⁰	1	1	1	0	0	0	1	1	1	0	0	0	0	0	1	1	1	2	0	1	1	1	1	1	0	0	0	0	0	2	0	2	2	0	1	1	4	1
S	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	3	2	4	3
T ¹	1	1	1	0	0	0	1	1	1	0	0	0	0	0	1	1	1	1	0	1	1	1	1	1	0	0	0	0	0	1	0	1	1	0	0	0	1	0
U ¹	1	1	1	0	0	0	1	1	1	0	0	0	0	0	1	1	1	2	0	0	1	1	1	1	0	0	0	0	0	2	0	2	2	0	1	1	4	1
T ²	1	1	1	0	0	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	1	1	1	1	0	0	0	0	0	1	0	1	1	0	0	0	1	0

<그림 13> 공급기업의 IMITAO 추론결과

세 번째 거래조건 <공급기업 →구매기업>

- 가격 : 표시된 정찰가격
- 구매수량 : 구매기업이 원하는 수량, 대량구입가능
- 운송기간 : 주문 후 7일
- 할인 : 대량구입시 가격할인 가능
- 대금지불 : 주문후 15일 이후에 지급
- 품질 : 품질인증 보장
- 환불정도 : 환불은 물건 수령 후 7일까지만 가능하고 100% 환불 가능
- 기타사항 : 구매기업의 정보에 대하여 보안시스템 제공

- C10 구매가격은 표시된 정찰가격입니까? Yes 1
 C15 구매수량은 일치합니까? Yes 1
 C9 구매물품은 품질인증을 획득하였습니까? Yes 1
 C14 운송기간은 주문 후 10일 이내입니까? Yes 1
 C5 추가적인 구매비용이 발생합니까? No 0
 C12 구매한 물품은 환불 및 반품이 가능합니까? Yes 1
 C11 구매대금은 주문 후 15일 이후에 지급합니까? Yes 1
 C25 구매하는 물품에 대하여 예산승인이 이루어졌습니까? Yes 1
 C22 공급기업과의 거래를 신뢰할 수 있습니까? Yes 1
 C26 거래시 구매기업의 정보에 대하여 보안을 유지할 수 있습니까? Yes 1

이를 구매기업의 추론메커니즘의 입력노드에 입력하면 그 추론결과가 다음 <그림 14>와 같이 제시된다.

<그림 14>의 결과에서 최종 진리상태인 T^2 에서 C27이 1이므로 구매기업이 공급기업이 수

정한 거래조건에 대하여 구매욕구가 높아졌음을 알 수 있다. 또한 거래 당사자인 공급기업의 경우에도 자신의 추론 메커니즘을 활용하여 현재 협상중인 구매기업과의 거래 성립여부를 확인하고자 할 것이다. 공급기업에서는 구매기업에 제시한 조건을 바탕으로하여 의사결정을 위하여 추론한 결과는 다음 <그림 15>와 같다.

이상과 같이 본 연구에서 제안하는 IMITAO는 기업간 전자상거래에서의 계약체결 협상과 같은 복잡한 협상과정에서도 유용하게 사용될 수 있음을 응용 사례를 통하여 증명하였다. 본 연구에서는 이미 서론에서도 언급한 바와 같이 전자상거래의 범위를 기업간(BtoB)의 경우로 제한하며, 이 경우 기업간 전자상거래에 있어서의 계약체결 문제는 구매기업과 공급기업이 다양한 조건을 고려해야 한다. 따라서, IMITAO의 빠른 추론능력과 실시간 문제해결 능력은 계약체결 문제이외에도 기업간 전자상거래의 다양한 협상문제에서도 유용하게 사용될 수 있을 것으로 기대된다.

4.4 상대적인 효율성 검증

지금까지 <그림 9>에 나타난 기업간 전자상거래 예제에 대하여 IMITAO를 사용하여 어떻게 기업간에 협상의사결정이 진행되는지를 살펴보았다. 본 절에서는 Lee & Cho(1999)에서

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22	C23	C24	C25	C26	C27	C28	C29	C30	C31	C32	C33
T0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	
M	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	5	3	1	1	7	2	2	3	2	5	7
W	0	0	0	0	0	0	0	0	1	1	1	1	2	1	1	2	0	2	0	2	0	1	5	4	1	1	9	0	1	4	2	5	9
T^0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	1	0	0	1	1	1	1
U^0	0	0	0	0	0	0	0	0	1	1	1	1	2	1	1	1	0	1	0	1	0	1	1	1	1	1	1	0	1	3	1	3	5
S	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	3	2	5	5
T^1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	1	0	0	1	0	0	1
U^1	0	0	0	0	0	0	0	0	1	1	1	1	2	1	1	1	0	0	0	1	0	1	0	1	1	1	1	0	1	3	1	3	5
T^2	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	0	1	0	1	1	1	1	0	0	1	0	0	1

<그림 14> 구매기업의 IMITAO 추론결과 - 구매욕구 높음

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22	C23	C24	C25	C26	C27	C28	C29	C30	C31	C32	C33	C34	C35	C36	C37	C38
T0	1	1	0	1	1	0	1	0	0	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
M	1	1	2	1	1	2	1	1	1	2	1	1	2	1	1	1	1	2	1	3	1	2	1	6	1	1	1	5	9	2	2	2	2	2	5	3	6	9
W	1	1	2	1	1	2	1	0	0	0	1	1	2	1	0	1	1	4	0	4	0	2	0	6	0	1	1	5	8	2	2	2	0	2	5	4	6	8
T ⁰	1	1	1	1	1	1	1	0	0	0	1	1	1	1	0	1	1	1	0	1	0	1	0	1	0	1	1	1	0	1	1	1	0	1	1	1	1	0
U ⁰	1	1	1	1	1	1	1	0	0	0	1	1	1	1	0	1	1	2	0	1	0	1	0	1	0	1	1	1	0	2	2	2	0	2	3	1	2	3
S	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	3	2	4	3
T ¹	1	1	1	1	1	1	1	0	0	0	1	1	1	1	0	1	1	1	0	1	0	1	0	1	0	1	1	1	0	1	1	1	0	1	1	0	0	1
U ¹	1	1	1	1	1	1	1	0	0	0	1	1	1	1	0	1	1	2	0	0	0	1	0	0	0	1	1	1	1	2	2	2	0	2	3	1	2	3
T ²	1	1	1	1	1	1	1	0	0	0	1	1	1	1	0	1	1	1	0	0	0	1	0	0	0	1	1	1	1	1	1	1	1	0	1	1	0	1

<그림 15> 공급기업의 IMITAO 추론결과 - 공급욕구 높음

제안된 추론 메커니즘과 본 연구에서 설명한 IMITAO의 추론 효율성을 서로 비교해 보고자 한다. 이를 위하여 먼저, 본 연구에서 제안하는 IMITAO 추론 메커니즘을 전체적으로 살펴볼 필요가 있다. 본 연구에서 제안하는 IMITAO 추론 메커니즘은 추론을 실시하기전에 지식 베이스를 구축하기 위해서 규칙행렬, 충분조건 벡터, 총경로행렬, 그리고 최소경로벡터 등을 미리 작성한다. 하지만 이러한 과정(규칙행렬, 총경로행렬, 최소경로벡터 계산)은 초기 진리상태 벡터로부터 시작되는 추론과정의 효율성과는 무관하다고 할 수 있다. 또한 최소경로란 의미 그대로 경로의 최소수를 의미하며 최단경로를 의미하는 것이 아니다. 이를 위해 본 연구에서 제안한 IMITAO가 Lee & Cho(1999)가 제안한 추론에 비하여 상대적으로 어느정도나 효율적인지를 알기 위하여 다음과 같은 통계량, 즉 효율성 지수(Efficiency Ratio)를 정의해보자.

$$ER = \frac{\text{Lee \& Cho(1999)를 통해 최중 진리상태벡터 } T^a \text{를 구하는데 필요한 계산량}}{\text{IMITAO를 통해 최중 진리상태벡터 } T^b \text{를 구하는데 필요한 계산량}}$$

따라서 ER의 값이 1보다 크면 클수록 Lee & Cho(1999)에 비해 IMITAO의 추론 효율성이 상대적으로 높다고 할 수 있다. 그런데 IMITAO 추론 과정에서 식 (6) 및 식 (7)을 이용하여 $\overline{T^k}$

로부터 $\overline{T^{k+1}}$ 을 구하는데 필요한 계산량은 Lee & Cho(1999)에서 제안된 추론 과정에서 식 (2) 및 식 (3)을 통해 T^k 로부터 T^{k+1} 을 구하는데 필요한 계산량과 같다. 다만 Lee & Cho(1999)에서 제안된 추론에 비해 IMITAO는 식 (4) 및 식 (5)를 이용하여 초기해를 구하는 추가적인 계산이 필요하다. 그런데 식 (4) 및 식 (5)에 필요한 계산량은 각각 식 (2) 및 식 (3)에 필요한 계산량과 같다. 따라서 a를 Lee & Cho(1999)에서 제안된 추론과정의 $T^a = T^F$ 를 만족하는 가장 작은 자연수라 하고, b를 IMITAO에서 $\overline{T^b} = \overline{T^F}$ 를 만족하는 가장 작은 자연수라 하면 이 경우 ER은 다음과 같이 계산된다.

$$ER = \frac{a}{b + 1}$$

Lee & Cho(1999)에서 제시된 추론 메커니즘을 활용하여 본 예제를 추론한 결과가 다음 <그림 16>과 같이 제시되어 있다.

<그림 16>의 결과를 살펴보면 Lee & Cho(1999)의 방법을 적용할 경우, 기업간에 최종적으로 협상이 타결되어 계약이 체결되는데 까지 필요한 계산량(T^a)이 구매기업의 경우에는 a=6이며, 공급기업의 경우에는 a=7인 것을 알 수 있다. 반면에 IMITAO를 적용하여 계약이 체결되기 위하여 필요한 계산량($\overline{T^b}$)은 <그림 15>에

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22	C23	C24	C25	C26	C27	C28	C29	C30	C31	C32	C33
S	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	3	2	5	5
T1	0	0	0	0	0	0	1	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	
U1	0	0	0	0	0	0	1	1	1	0	0	1	1	1	1	0	0	0	0	0	0	1	0	0	1	1	0	0	1	2	0	1	3
T2	0	0	0	0	0	0	1	1	1	0	0	1	1	1	1	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0
U2	0	0	0	0	0	0	1	2	1	0	0	1	2	1	1	1	0	0	0	0	0	1	0	0	1	1	0	0	2	3	0	1	3
T3	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	0	0	0	0	0	1	0	0	1	1	0	0	1	1	0	0	0
U3	0	0	0	0	0	0	1	2	1	0	0	1	2	1	1	2	0	0	0	1	0	1	0	1	1	1	0	0	3	4	1	2	3
T4	0	0	0	0	0	0	1	1	1	0	0	1	1	1	1	1	0	0	0	1	0	1	0	1	1	1	0	0	1	1	0	0	0
U4	0	0	0	0	0	0	1	2	1	0	0	1	2	1	1	2	0	0	0	2	0	1	0	2	1	1	0	0	3	4	1	3	5
T5	0	0	0	0	0	0	1	1	1	0	0	1	1	1	1	1	0	0	0	1	0	1	0	1	1	1	0	0	1	1	0	0	1
U5	0	0	0	0	0	0	1	2	1	0	0	1	2	1	1	2	0	0	0	2	0	1	0	2	1	1	1	0	3	4	1	3	6
T6	0	0	0	0	0	0	1	1	1	0	0	1	1	1	1	1	0	0	0	1	0	1	0	1	1	1	1	0	1	1	0	0	1

(a) 구매기업의 추론결과 - 구매욕구높음

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22	C23	C24	C25	C26	C27	C28	C29	C30	C31	C32	C33	C34	C35	C36	C37	C38	
S	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	3	2	4	3			
T1	1	1	0	1	1	0	1	0	0	1	1	0	0	1	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0		
U1	1	1	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	1	0	0	0	1	1	0	0	2	2	1	0	1	1	0	1	0	
T2	1	1	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	1	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0		
U2	1	1	1	1	1	1	1	0	0	1	1	0	0	1	0	2	1	0	0	0	1	2	0	0	0	1	2	0	0	3	3	2	0	1	1	0	2	2	
T3	1	1	1	1	1	1	1	0	0	1	1	0	0	1	0	1	1	0	0	0	1	1	0	0	0	1	1	0	0	1	1	1	0	0	0	0	0	0	
U3	1	1	2	1	1	2	1	0	0	1	1	0	0	1	0	2	1	2	0	0	1	2	0	0	0	1	2	0	0	3	3	3	0	1	3	0	2	2	
T4	1	1	1	1	1	1	1	0	0	1	1	0	0	1	0	1	1	1	0	0	1	1	0	0	0	1	1	0	0	1	1	1	0	0	1	0	0	0	
U4	1	1	2	1	1	2	1	0	0	1	1	0	0	1	0	2	1	3	0	0	1	2	0	0	0	1	2	1	0	3	3	3	0	1	4	1	3	2	
T5	1	1	1	1	1	1	1	0	0	1	1	0	0	1	0	1	1	1	0	0	1	1	0	0	0	1	1	1	0	1	1	1	0	0	1	0	0	0	
U5	1	1	2	1	1	2	1	0	0	1	1	0	0	1	0	2	1	3	0	0	1	2	0	0	0	1	2	2	0	3	3	3	0	1	4	1	3	3	
T6	1	1	1	1	1	1	1	0	0	1	1	0	0	1	0	1	1	1	0	0	1	1	0	0	0	1	1	1	0	1	1	1	0	0	1	0	0	1	
U6	1	1	2	1	1	2	1	0	0	1	1	0	0	1	0	2	1	3	0	0	1	2	0	0	0	1	2	2	1	3	3	3	0	1	4	1	3	4	
T7	1	1	1	1	1	1	1	0	0	1	1	0	0	1	0	1	1	1	0	0	1	1	0	0	0	1	1	1	1	1	1	1	1	0	0	1	0	0	1

(b) 공급기업의 추론결과 - 공급욕구 높음

<그림 16> Lee & Cho (1999) 추론방법을 이용한 최종추론결과

제시되어 있듯이 구매기업 공급기업 모두 $b=2$ 이다. 따라서 Lee & Cho(1999)와 비교하여 IMITAO가 갖는 추론상의 효율성 지수인 ER은 다음 <표 2>와 같이 제시할 수 있다.

<표 2>를 보면 본 연구에서 제시한 기업간 전자상거래 협상의사결정 지원을 위한 추론 메커니즘인 IMITAO가 Lee & Cho(1999)에서 제안된 추론메커니즘과 비교하여 갖는 효율성은 2배 이상 높은 것을 알 수 있다.

<표 2> [그림 9]에서 제시된 사례에 대한 IMITAO의 ER(Lee & Cho(1999)와 비교하여)

루트노드		최종 진리상태벡터 T^F		ER
		Lee & Cho (1999)	IMITAO	
구매/공급요구 높음 (계약체결)	구매 기업	T^6	$\overline{T^2}$	$2.0(\frac{6}{(1+2)})$
	공급 기업	T^7	$\overline{T^2}$	$2.5(\frac{7}{(1+2)})$

V. 결 론

본 연구논문에서는 기업간 전자상거래에서 필요한 다양한 협상지원 추론메카니즘 설계를 위하여 기존의 전문가시스템 영역에서 지식표현 및 추론도구로 널리 활용되어 온 AND-OR 그래프를 변환하여 이를 TAOG라는 이름으로 새롭게 제안하였다. TAOG에서는 복합노드 개념을 도입하여 복합규칙이 갖는 전제조건에 참 또는 거짓 여부를 정수제산화 하므로써 복잡한 기업간 전자상거래 추론과정을 보다 신속하고 정교하게 만들 수 있는 여지를 마련하였다.

이러한 TAOG의 장점을 확인한 후 본 연구에서는 여기에 규칙행렬, 충분조건 벡터, 총경로행

렬 등의 개념을 적용하여 정수행렬연산에 기초한 새로운 기업간 전자상거래 추론 메커니즘인 IMITAO를 제안하고 그 성과를 실제문제를 토대로 하여 검증하였다. 또한 기존 연구에서 제안한 추론방법과 비교하는 실험도 하므로써 본 연구에서 제안한 IMITAO가 약 2배 이상 더 효율적인 추론을 할 수 있음을 검증하였다.

그러나, 향후 연구방향으로는 사례기반추론(CBR)과 같은 다른 지능형 기법과의 결합을 통하여 본 연구에서 제안한 IMITAO의 추론능력을 더욱 확장하는 작업이 필요하다. 아울러, 이러한 연구결과를 더욱 확장하여 보다 복잡한 거래조건 협상상황에서도 효과적으로 기업간 전자상거래 당사간의 거래조건 최적화에 적용될 수 있도록 하는 것이 필요하다.

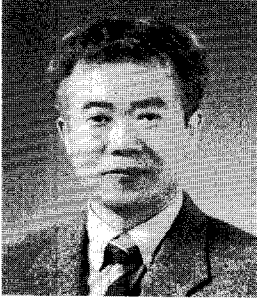
〈참 고 문 헌〉

- [1] Buchanan, B.G., and Shortliffe, E.H., *Rule-Based Expert Systems*. Reading, Mass. Addison-Wesley, 1984
- [2] Chavez, A. and Maes, P., "Kasbah: An Agent Marketplace for Buying and Selling Goods," *The First International Conference and Exhibition on the Practical Application of Intelligent Agent and Multi-Agent Technology '96*, 75-90, 1996.
- [3] Dennis, A.R., "Lessons from Three Years of Web Development," *Communications of the ACM*, 41(7), 112-113, 1998.
- [4] Giarratano, J. and Riley, G., *Expert Systems: Principles and Programming*. PWS Publishing Company, 1994.
- [5] Hoffman, D.L. and Novak, T.P., "Marketing in Hypermedia Computer-Mediated Environments: Conceptual Foundations," *Journal of Marketing*, 60, 50-68, 1996.
- [6] Jarvenpaa, S.L. and Todd, P.A., "Consumer Reactions to Electronic Shopping on the World Wide Web," *International Journal of Electronic Commerce*, 1(2), 59-88, 1997.
- [7] Kim, J.W., "Metaphoric Navigation Aids for Cyber Shopping Malls: An Empirical Study Toward the Construction of Consumer Interface," *International Journal of Management Science*, 3(2), 1-27, 1997.
- [8] Lederer, A.L., D.A. Mirchandani, and K. Sims, "Electronic Commerce: A Strategic Application?," *SIGCPR/SIGMIS'96*, 277-287, 1996.
- [9] _____, Mirchandani, and K. Sims, "The Link Between Information Strategy and Electronic Commerce," *Journal of Organizational Computing and Electronic Commerce*, 7(1), 17-34, 1997.
- [10] Lee, K.C and H.R. Cho, "An Extended AND-OR Graph-based Simulation and

- Electronic Commerce," *Simulation Proceeding of Korean Society for Simulation Conference*, 242-250, 1999.
- [11] Lohse, G.L. and Spiller, P., "Electronic Shopping," *Communications of the ACM*, 41(7), 81-87, 1998.
- [12] O'Connor, G. and O'Keefe, B., "Viewing the Web as a Marketplace: The Case of Small Companies," *Decision Support Systems*, 21, 171-183, 1997.
- [13] O'Keefe, R.M and Mceachern, T., "Web-based Customer Decision Support Systems," *Communications of the ACM*, 41(3), 71-78, March 1998.
- [14] Rebello, K., Armstrong, L., and Cortese, A., "Making Money on the Net," *Business Week*, September, 23, 1996.
- [15] Schrooten, R., "Agent-based Electronic Consumer Catalogs," *The First International Conference and Exhibition on the Practical Application of Intelligent Agent and Multi-Agent Technology'96*, 543-571, 1996.
- [16] Shaw, M.J., Gadner, D.M, and Thomas, H, "Research Opportunities in Electronic Commerce," *Decision Support Systems*, 21, 149-156, 1997.
- [16] Spiller, P. and G.L. Lohse, "A Classification of Internet Retail Stores," *International Journal of Electronic Commerce*, 2(2), 29-56, 1998.
- [17] Westland, J.C. and Clark, T.H., *Global Electronic Commerce: Theory and Case Studies*. Massachusetts Institute of Technology, 1999.

◆ 이 논문은 1999년 9월 13일 접수하여 1차 수정을 거쳐 2000년 12월 16일 게재 확정되었습니다.

◆ 저자소개 ◆



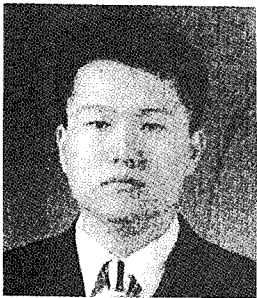
이건창 (Lee, Kun Chang)

현재 성균관대학교 경영학부 교수로 재직 중이다. 성균관대학교 경영학과를 졸업하고, 한국과학기술원(KAIST) 경영과학과에서 경영정보시스템 전공으로 석사 및 박사학위를 취득하였다. 주요 관심분야는 전자상거래, 퍼지인식도, 협상지원시스템, 지식경영, 인터넷 마케팅 등이다.



조형래 (Cho, Hyung Rae)

서울대학교 계산통계학과에서 학사(1983), 그리고 한국과학기술원 경영과학과에서 박사학위(1989)를 취득하고, 현재 경상대학교 산업시스템공학부에서 부교수로 재직중이다. 주요 연구분야로는 경영정보시스템, 전문가시스템, 전자상거래, 정보경제학 등이다



권순재 (Kwon, Soon Jae)

성균관대학교 경영학부(1998)를 졸업하고 성균관대학교 경영학과(2000)에서 석사학위를 취득한 후, 현재 성균관대학교 경영학부 박사과정에 재학 중이다. 주요 관심분야로는 인공지능기법을 이용한 의사결정, 인터넷에서의 소비자행동, 지식경영, 퍼지인식도, 협상지원시스템, 웹로그 분석을 통한 소비자 개인화, 전자상거래 등이다.