

# DSS와 사례기반 추론의 결합

김진백\*

## 〈 목 차 〉

- |                         |                         |
|-------------------------|-------------------------|
| I. 서론                   | IV. DSS와 사례기반 추론의 결합    |
| II. DSS의 구조와 의사결정과정     | 1. DSS와 사례기반 추론의 결합필요성  |
| 1. DSS의 구조              | 2. 의사결정지원 도구로서의 사례기반 추론 |
| 2. 의사결정과정과 지원도구         | 3. 사례기반 DSS의 구조         |
| III. 사례와 사례기반 추론        | 4. 사례기반 DSS의 구성요소       |
| 1. 사례와 사례표현             | V. 결론                   |
| 2. 사례기반 추론              | 참고문헌                    |
| 3. 사례기반 추론과 유사추론 방법의 비교 | Abstract                |

## I. 서론

의사결정이란 기업경영에 있어서 중요한 임무중 하나로 인식되고 있다. 관리자 들은 업무수행과정에서 많은 의사결정을 내리게 되며 그 결과가 효력을 발휘하기 위해서는 시간적 제약조건과 정확성 제약조건을 만족시켜야 한다. 최근 경영환경의 변화속도가 증가함에 따라 의사결정에 대한 시간적 제약조건이 더욱 엄격해지면서 정확성 제약조건이 미충족될 가능성이 커지고 있다. 정보기술의 활용은 엄격해진 시간적 제약조건과 정확성 제약조건을 보다 충실히 만족시켜주기 위한 대안으로 이용될 수 있다.

현재 정보기술을 활용한 다양한 형태의 의사결정지원 도구들이 제시되었으며 이 들중 대표적인 것이 의사결정지원시스템(Decision Support Systems : DSS)이다. 그러나 DSS는 모형구축과정이 어려워 의사결정자가 이를 직접 사용하지 못하고 전문스텝의 도움을 받아 사용되는 것이 일반적인 사용방식으로 되어 있다(Turban, 1990). 이로 인해 DSS에

\*부산여자대학 경영정보과

의한 해결안의 타당성 및 수용도가 저하되는 문제가 발생하게 된다. 이러한 문제는 기존의 DSS에서 사용자가 수행하는 모형구축과정을 시스템이 수행하게 함으로써 해결이 가능하다. 이를 위해 최근에는 문제영역에 대한 지식을 표현하기 위해 DSS와 인공지능(Artificial Intelligence : AI) 분야의 기술이 결합된 지적 의사결정지원시스템(Intelligent DSS)에 대한 필요성이 증대되고 있다.

AI 시스템은 많은 지식을 필요로 하는 지식집약적 시스템(knowledge-intensive system)으로 시스템의 지식은 인간 혹은 정보시스템의 자체 학습기능에 의해 형성된다. 전문가시스템은 전자의 방법에 의해 구축되는 대표적인 시스템으로서 시스템이 이용하는 지식은 생성규칙(production rule)의 형태로 저장되는 것이 일반적이다. 후자의 방법은 전문가로부터 지식획득이 어렵거나 지식획득과정을 자동화하기 위해 이용되는 방법으로 이에 의한 대표적인 시스템이 사례기반 추론시스템(Case-Based Reasoning Systems : CBRS)이다.

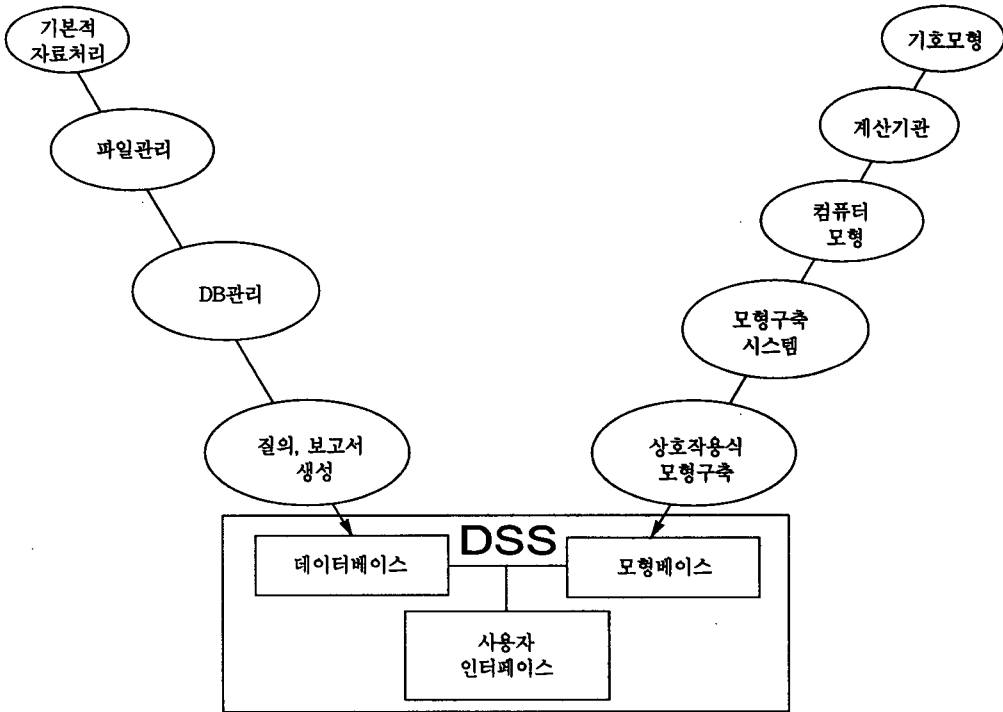
CBRS는 과거의 의사결정 경험을 이용해서 당면 문제에 대한 해결책을 보다 신속히 제시하려는 시스템으로서 기존의 전문가시스템보다 구현과 유지보수 측면 등에서 많은 이점을 가지고 있다. 또한 전문가시스템의 주요 지식표현수단인 규칙은 도메인 지식의 풍부함과 미세한 차이를 표현하기에는 부적합하기 때문에 전문가시스템보다 CBRS에 대한 선호도가 더 크다고 할 수 있다(Gupta, 1994). 그리고 CBRS는 지식획득과정에 대한 병목현상을 해결해줄 수 있기 때문에 성장잠재력이 다른 시스템에 비해 매우 클 것이라 예상되고 있다(Mott, 1993).

본 논문에서는 효율적인 의사결정지원을 위해 기존의 DSS에 AI 분야의 사례기반 추론(Case-Based Reasoning : CBR) 기법을 도입하기로 한다. CBR 기법은 과거의 문제해결 상황과 해결방법을 사례(case) 형식으로 저장해두고 이를 새로운 문제해결에 재이용하려는 것이다. CBR 기법을 이용할 경우 모형구축단계는 기존의 모형을 재이용하는 것으로 대체될 수 있으므로 사용자들의 의사결정 부담을 줄일 수 있다. 그러나 DSS와 CBR 기법은 각각의 기본틀이 매우 상이하기 때문에 결합방식에 따라 효율성에 많은 차이를 나타내게 된다. 따라서 본 논문에서는 의사결정지원의 효율성 제고를 위해 양자의 결합에 따른 시스템의 구조문제를 알아보기로 한다.

## II. DSS의 구조와 의사결정과정

### 1. DSS의 구조

정보시스템의 궁극적인 목적은 조직의 목표달성을 지원하는 것으로 이를 위해 다양한 정보시스템들이 존재한다. 그 중 DSS는 신속한 의사결정을 가능하게 하거나 의사결정의 질을 향상시킴으로써 조직의 목표달성을 지원하게 된다. DSS는 데이터베이스, 모형베이스, 사용자 인터페이스를 기본 구성요소로 하며, 데이터 질의와 상호작용식 모형구축중 어디에 비중을 두느냐에 따라 데이터지향형(data oriented) DSS와 모형지향형(model oriented) DSS로 분류될 수 있다(Alter, 1977). 이는 Sprague(1987)의 주장과 유사한 것으로, 그에 따르면 DSS의 출현동기는 데이터처리분야와 모형구축을 위한 경영과학분야의 발전이 교차하였기 때문이라는 것이다.



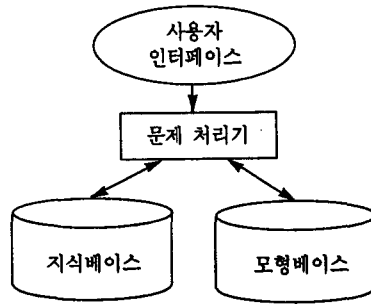
〈그림 1〉 DSS의 진화과정 및 구성요소

그러므로 DSS는 <그림 1>과 같이 자료처리 분야와 모형구축 분야가 각각의 발전단계를 따라 진화하다가 양자의 상호보완성으로 인해 결합된 정보시스템이라 할 수 있다. 자료처리 분야의 발전은 자료를 분류, 정렬, 요약하는 기본적 자료처리단계에서 진화되어 점차 임의 질의(ad hoc query)방식에 대응하기 위한 질의 및 보고서 생성단계로 발전하였다. 그러나 데이터 지향형 DSS는 Simon의 문제해결 3단계중 정찰(intelligence) 단계만을 주로 지원해줌으로써 설계(design)와 선택(choice) 단계를 추가로 지원하기 위해서 모형지향형 DSS의 기능을 보강할 필요가 있다.

모형구축분야는 선형/비선형 방정식에 의한 기호모형을 이용하는 단계에서 시작해서 계수추정이나 방정식 계산에 컴퓨터를 처음으로 이용하게 되는 계산기관(computational engine) 단계로 발전하였다. 그후 프로그램을 컴퓨터 모형으로 취급하는 3단계를 거쳐 모형구축을 직접 지원하는 모형구축 시스템 단계와 상호작용식 모형구축 단계로 진화하였다.

이러한 출현동기로 인해 초기의 DSS들은 주로 데이터베이스의 관점에서 관계형 DB 접근법에 따라 모형들을 통합하려고 시도하였다. 그러나 최근에는 단순한 데이터처리기능으로 복잡한 모형을 다루기가 어렵다는 점을 인식하여 모형관리의 관점에서 데이터부분의 통합을 시도하고 있다. 그러나 모형베이스는 경영과학분야와 정보기술분야가 결합되어 생성된 부분이기 때문에 모형베이스가 중심이 되는 DSS의 경우에는 모형구축기능에 중점을 둠으로써 의사결정자들이 시스템 이용시 전문스태프의 도움을 받아야만 이용이 가능하다는 문제점이 있다.

이러한 문제점을 극복하기 위해서 최근에는 DSS에 지능을 부여한 지적 의사결정지원 시스템을 개발하려는 연구가 진행되고 있다. 전통적 의사결정지원시스템과 달리 지적 의사결정지원시스템은 데이터뿐만 아니라 의사결정에 필요한 지식도 포함함으로써 필요한 모형을 시스템이 모형베이스에서 자동선택할 수 있어야 하며, 이는 문제처리기(problem processor)라는 구성요소를 통해 이루어진다. 문제처리기는 입력된 문제를 해결하기 위해 적절한 전략을 선택해야 하며, 이는 저장된 지식들과 상호작용을 할 수 있는 형식으로 질의를 변화시킴으로써 관련된 작업들이 수행된다. 따라서 이러한 기능을 가진 지적 의사결정지원시스템의 구조는 <그림 2>와 같이 나타낼 수 있다(Fedorowicz & Williams, 1986).



〈그림 2〉 지적 의사결정지원시스템의 일반적 구조

## 2. 의사결정과정과 지원도구

의사결정이란 목표달성을 위해 행위 대안들중 하나를 선택하는 과정이다. Simon에 따르면 의사결정과정은 정찰(Intelligence), 설계(Design), 그리고 선택(Choice)이라는 3단계에 걸쳐 이루어진다. 정찰이란 의사결정의 필요성을 인식하고 의사결정에 필요한 데이터를 수집하는 단계이다. 의사결정의 필요성은 문제와 기회를 발견함으로써 발생한다. 문제의 발견이란 현재의 업무수행결과가 계획과 차이가 있거나 예상하지 못한 위기 상황이 발생하였음을 감지하는 것이다. 기회의 발견이란 의사결정자가 보다 유리한 결과를 얻을 수 있는 상황을 감지하는 것이다. 따라서 문제가 발생하면 이를 해결하기 위해 의사결정이 행해지며, 기회가 발생하면 이를 달성하기 위해 의사결정이 행해진다. 따라서 정찰단계는 주로 데이터지향형 DSS에 의해 지원되고 있다(Liang, 1985).

설계단계는 대안을 생성하는 단계로서, 생성된 대안들은 정성적 또는 정량적 분석기법을 이용해서 가능한 결과들을 예측하기 위해 모형의 형태로 수립된다. DSS는 다른 정보시스템들보다 경영과학분야의 영향을 많이 받았기 때문에 대부분 계량적인 모형으로 대안들을 표현하게 된다. 의사결정과정에 모형을 이용하는 이유는 비용과 시간을 절약하고 실험을 용이하게 하기 위해서이다. 의사결정과정중 시간과 비용이 가장 많이 소요되는 부분은 모형수립부분이다. 기존의 DSS들은 모형수립과정의 대부분을 사용자에게 의존하고 있다. 즉, 기존의 DSS는 설계단계의 주요 활동인 대안의 생성 및 모형구축부분에 대한 지원이 미흡하다. 이로 인해 DSS는 의사결정과정을 수동적으로 지원하는 역할을 수행한다. 최근 DSS의 연구분야중 모형관리시스템(Model Management Systems)을 중심으로 한 연구에서는 모형수립의 자동화를 위해 모형관리시스템의 기능에 AI 시스템의 기

능을 결합해서 지적 모형관리시스템을 구축하려는 연구가 진행되고 있다.

선택단계는 설계단계에서 수립한 대안들을 평가하는 단계로서 각 대안별 예측치를 도출하는 단계이다. 모형지향형 DSS는 모형들을 통해 가능한 결과의 예측에 이용될 수 있다. 그러나 대부분의 모형지향형 DSS는 정성적 분석모형이 부족하기 때문에 계량화할 수 없는 문제에 대한 결과의 예측에는 유용성이 매우 낮다. 따라서 현재 모형지향형 DSS는 의사결정자가 수립한 계량적 모형의 결과 예측만을 지원해준다.

AI 시스템은 DSS와 달리 문제해결과정을 시스템이 주도해나간다. 따라서 문제해결에 필요한 대안의 생성과 최적안의 결정은 AI 시스템이 수행하는 주요 과업들이 된다. 즉, AI 시스템은 정성적 모형을 통해 의사결정단계중 설계와 선택의 단계를 중점적으로 지원한다고 볼 수 있다. 따라서 의사결정단계별 지원도구는 <표 1>과 같다.

<표 1> 의사결정단계별 지원도구

의사결정단계	의사결정 지원도구
정찰	데이터지향형 DSS
설계	AI 시스템
선택	모형지향형 DSS, AI 시스템

DSS는 데이터베이스와 모형베이스를 구성요소로 하기 때문에 의사결정과정중 정찰과 선택단계를 효과적으로 지원해줄 수 있는 시스템이다. 그리고 AI 시스템은 지식베이스에 사실적인 자료와 정성적인 절차모형을 가지고 있으므로 추론기관을 통해 의사결정단계중 설계와 선택단계를 효과적으로 지원해줄 수 있다.

### Ⅲ. 사례와 사례기반 추론

#### 1. 사례와 사례표현

사례(case)란 문제해결경험에 대한 지식을 표현하는 단위로서 구성과 추론단계에 따라 여러 종류로 분류될 수 있다. 일반적으로 사례라고 하면 사례저장소에 저장되어 있는 사례를 의미하며, 이를 저장된 사례(stored case) 혹은 기존 사례(existing case)라고 한다. 저장된 사례는 도메인 종속사례(domain dependent case)와 도메인 독립사례(domain

independent case)로 나뉠 수 있다. 도메인 종속사례는 경험적 지식을 사례형식으로 표현해 놓은 것이다. 그리고 도메인 독립사례는 경험적 지식 대신 일반지식을 사례형식으로 표현해 놓은 것이다. CBR은 과거의 경험을 이용하는 추론방법이므로 대부분의 CBRS들은 도메인 종속사례만을 이용한다. 또한 도메인 종속 혹은 독립사례들은 사례의 구성에 따라 단순사례(simple case)와 복합사례(complex case)로 나뉠 수 있다. 단순사례란 다른 사례를 자신의 일부로 갖지 않는 사례를 의미하며, 복합사례는 자신의 일부로 다른 사례를 갖는 경우를 의미한다. CBR 과정에서 문제해결에 가장 적합한 사례로 선택된 사례는 기저사례(base case)라고 부른다. 그리고 수정된 사례(adapted case)는 문제해결을 위해 변형된 기저사례를 말하며, 사례검색을 위해 현 문제를 사례형식으로 표현한 경우 이를 새로운 사례(new case)라고 한다.

다양한 형태의 사례들이 존재하지만 이들은 상호 매우 유사하다고 볼 수 있다. 현재 이들 사례내에 정확히 어떤 정보가 포함되어야 하는가에 대해서는 아직 합의가 이루어지지 않은 상태이다. 그 이유는 첫째, CBR에 대한 연구동기가 2가지 방향, 즉 인간의 지식집약적 업무수행을 모형화하려는 의도와 기존의 데이터베이스나 지식저장소를 이용하는 의도에서 나왔기 때문이다. 둘째는 CBR의 다양한 업무영역이 사례표현에 대해 다양한 기능요구사항을 부과하기 때문이다(Alterman, 1989). 그러나 일반적으로 사례는 문제기술 부분과 해결안 부분을 주요 구성요소로 갖는다(Mahapatra & Sen, 1994). 문제기술 부분은 의사결정과 관련된 문제의 속성에 대한 정보를 가지며, 해결안 부분은 의사결정이나 문제해결에 필요한 해결절차에 대한 정보를 가지고 있다.

문제기술 부분의 내용은 사례의 미래 용도와 새로운 문제해결 상황과의 비교를 위해 어떤 정보가 필요한가에 따라 다르다. 일반적으로 문제기술 부분에 어떤 기술적 정보(descriptive information)를 포함할 것인가에 대한 결정은 <표 2>의 두가지 일반규칙을 따르면 된다(Kolodner, 1993).

<표 2> 문제기술 부분에 포함될 정보의 결정규칙

규칙 1	목표달성을 위해 명확히 고려 대상이 되는 모든 정보를 포함해야 한다.
규칙 2	일반적으로 해당 유형의 사례를 기술할 때 이용되는 정보를 포함해야 한다.

특히 기술적 정보의 포함 여부에 대한 결정규칙은 이론적 기반이 취약한 영역에 있어서 중요시 되며, 규칙 1은 명확히 관련된 것들은 문제기술 부분에 포함되어야 한다는 것

을 의미한다. 따라서 관련은 되나 변동이 전혀 없는 묵시적인 속성들에 대해서는 무시할 수 있다. 규칙 2는 접근가능성을 위해 필요한 규칙이다. 일반적으로 과거의 사례에 대한 기술내용이 새로운 문제에 대한 기술내용과 유사할수록 검색과 비교는 더욱 쉬워진다. 따라서 새로운 문제를 해결하기 위한 정보나 새로운 상황을 해석하기 위한 정보는 문제 기술 부분에 포함되어야 한다.

해결안 부분의 내용은 시스템의 응용분야에 따라서 매우 다양한 형태가 된다. 설계시스템에 있어서 해결안 부분은 설계한 인공조형물이 되며, 계획시스템에 있어서 해결안 부분은 도출된 계획이 된다. 또한 해석시스템에 있어서 해결안 부분은 해석이나 분류 결과로 되며, 설명시스템에 있어서는 설명 그 자체가 해결안이 된다. 따라서 각 사례에 기술되어 있는 해결안 부분은 문제기술 부분에 기술되어 있는 목표의 달성이나 제약조건을 충족시켜 줄 수 있는 개념이나 객체들이라 볼 수 있다.

2. 사례기반 추론

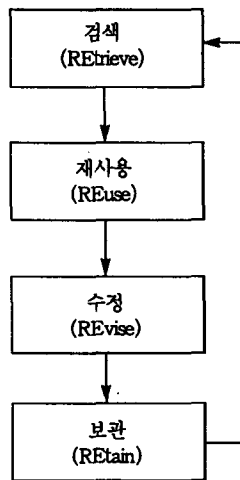
사례기반 추론(Case-Based Reasoning : CBR)은 새로운 문제가 주어지면 이를 분석해서 과거의 경험중 가장 유사한 경험을 검색한 후, 검색된 과거 경험을 직접 혹은 부분적인 수정을 통해 문제해결에 이용하는 과정을 의미한다. 사례기반 추론의 과정은 학자에 따라 <표 3>과 같이 여러 단계로 세분화되고 있다.

<표 3> 사례기반 추론과정의 분류 형태

Aamodt 등	Slade	McGovern 등
1. 검색	1. 색인부여 2. 검색	1. 문제의 명세화 2. 관련 사례의 검색
2. 재사용	3. 수정 4. 테스트	3. 최적 사례의 선택 4. 해결안 구축 5. 해결안 실험 6. 해결안 시행
3. 수정	5. 실패에 대한 설명 및 수정	7. 해결안 평가
4. 보관	6. 저장	8. 사례 기억장소 갱신



Slade(1991)는 CBR 과정을 과정모형(process model)을 통해 새로운 문제의 속성에 대한 색인부여, 검색, 수정, 테스트 및 저장이라는 5 단계로 나눈 후 실패에 대한 설명 단계를 추가해서 총 6 단계로 나누었다. McGovern 등(1994)은 CBR 과정을 문제의 명세화, 검색, 선택, 해결안의 구축, 해결안의 실험, 해결안의 시행, 해결안의 평가 및 사례 기억장소의 갱신이라는 8단계로 나누었다. Aamodt 등(1994)은 CBR 과정을 검색, 재사용, 수정, 보관이라는 4단계로 나누었다. CBR의 과정은 학자에 따라 세분화의 정도에 차이는 있으나 상호 매우 유사한 과정으로 나타나고 있다. 즉, Slade의 색인부여와 검색단계, McGovern 등의 문제의 명세화와 관련 사례의 검색단계는 Aamodt 등의 검색단계에 해당한다. 또한 Slade의 수정과 테스트 단계, McGovern 등의 3단계 최적 사례의 선택단계부터 6단계 해결안 시행단계까지는 Aamodt 등의 재사용단계에 해당한다. Slade의 실패에 대한 설명 및 수정단계, McGovern 등의 해결안 평가단계는 Aamodt 등의 수정단계에 해당된다. 그리고 Slade의 저장단계와 McGovern 등의 사례 기억장소 갱신단계는 Aamodt 등의 보관단계에 해당된다. 따라서 이들중 현재 많이 이용되고 있는 Aamodt 등이 나눈 4단계 모형인 4REs를 통해 CBR의 과정을 살펴보기로 한다.



<그림 3> 사례기반형 추론과정(4 REs)

- ① 검색(REtrieve) : 새로운 문제가 주어지면 CBR에서는 문제의 주요 속성을 추출한 다음 이들 속성을 이용해서 과거의 유사한 사례를 검색하게 된다. 검색결과는 3가

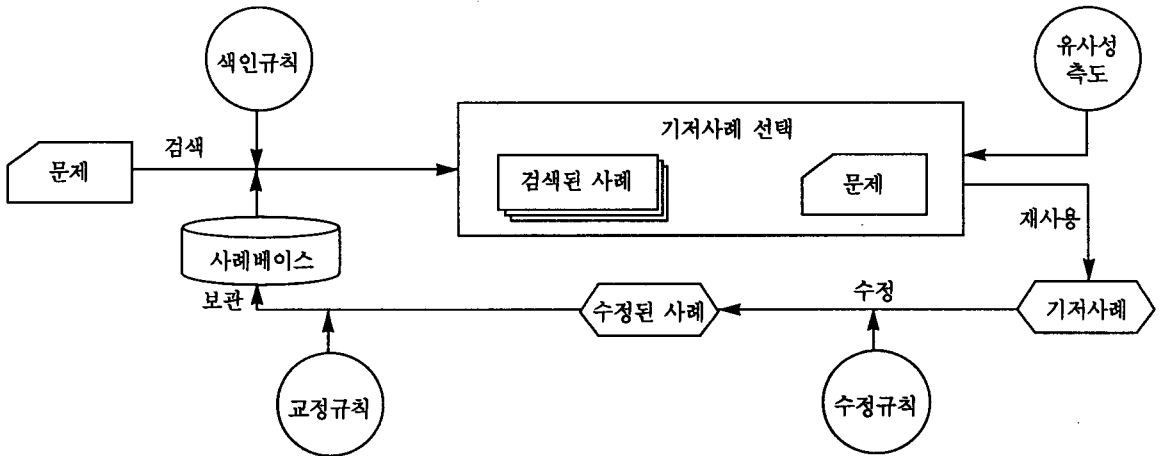
지로 요약될 수 있으며, 첫째는 둘 이상의 유사한 사례가 검색되는 경우, 둘째는 유일한 유사 사례가 검색되는 경우, 셋째는 유사한 사례가 발견되지 않을 경우이다.

- ② 재사용(REUse) : 1단계의 검색결과 단일 사례가 검색되면 이를 통해 문제를 해결하게 되며 둘 이상의 유사한 사례가 검색되면 속성별로 비교를 통해 유사성을 측정한 후 주어진 문제와 가장 유사성 측정치가 높은 사례를 기저사례로 선택해서 새로운 문제의 해결을 시도한다.
- ③ 수정(REvise) : 2단계를 통해 선택된 기저사례가 새로운 문제와 완전히 일치할 수도 있으나 경우에 따라서는 부분적인 차이가 발생할 수도 있다. 따라서 이 경우에는 사용자와의 상호작용을 통해서 차이부분에 대한 부분적인 수정과정을 거쳐서 새로운 문제에 적합한 해를 생성시키게 된다.
- ④ 보관(REtain) : 새로운 문제가 기존의 사례와 완전히 일치하였을 경우 새로운 문제의 해결방법을 저장하면 중복저장이 되므로, 이 경우에는 새로운 문제의 해결방법을 저장할 필요가 없다. 그러나 새로운 문제가 기존의 사례와 차이가 있을 경우에는 새로운 문제의 해를 사례베이스에 저장해서 향후 의사결정에 이용할 수 있게 한다.

이상의 4 REs 과정은 주기성을 띠는 것으로서 대부분의 시스템들은 CBR 과정을 지원하기 위해 색인규칙(indexing rules), 사례베이스(CaseBase : CB), 유사성 측도(similarity metrics), 수정규칙(modification rules), 교정규칙(repair rules) 등과 같은 지식베이스들이 이용된다.

〈그림 4〉는 4 REs와 이상의 다섯 가지 지식들을 이용해서 새로운 문제를 해결하는 CBR 과정으로서 이는 문제발생시마다 반복적으로 발생하는 순환시스템의 성격을 가진다. 순환시스템의 첫번째 단계는 문제가 입력되면 이의 주요 속성에 대한 색인을 이용해서 사례베이스를 검색하는 단계이다. 두번째 단계는 사례베이스의 검색결과와 문제를 비교하는 단계로서 이때 유사성 측도를 이용해서 검색된 사례들중 가장 유사성 측도가 높은 사례를 기저사례로 선택하며, 선택된 기저사례는 문제해결을 위해 재사용된다. 세번째 단계는 기저사례로 선택된 기존 사례가 새로운 문제에 대한 사례와 완전히 일치하지 않을 경우 수정규칙을 이용해서 기존 사례의 내용을 수정해서 문제에 적합한 해결안을

생성하는 단계로서 이때 수정된 사례가 생성된다. 네번째 단계는 문제가 기존의 사례와 상이하었을 경우 교정규칙을 이용해서 사례베이스에 새로운 사례로 저장하는 단계이다.



〈그림 4〉 사례기반 추론의 순환과정

### 3. 사례기반 추론과 유사추론 방법의 비교

CBR이란 용어가 유사 추론방법들에 대한 일반적인 용어로 이용되지만 고유의 CBR은 몇 가지 측면에서 유사 추론방법들과 구별이 가능하다. 첫째, 전형적인 사례는 해당 사례내에 일정한 수준의 풍부한 정보를 가지고 있으며 사례의 내부 조직측면에서 다소의 복잡성을 갖는다. 즉, 일부 값을 갖는 속성 벡터는 전형적 사례의 기술내용이 아니다. 둘째, 고유의 CBR은 다양한 문제해결상황에 응용시 검색된 해를 수정할 수 있다는 특징이 있다. 셋째, 고유의 CBR은 일반적인 배경지식을 활용한다는 특징이 있다. 그러나 CBR 패러다임의 범위를 과거의 경험에 관련된 지식을 조직, 검색, 활용, 색인화하기 위한 여러 방법들을 포괄하는 것으로 할 경우 전형기반 추론(Exemplar-Based Reasoning : EBR), 예기반 추론(Instance-Based Reasoning : IBR), 기억기반 추론(Memory-Based Reasoning : MBR), 상사기반 추론(Analogy-Based Reasoning : ABR) 등도 CBR의 패러다임에 속하는 것으로 볼 수 있다.

이들 중 CBR과 동의어로서 가장 많이 사용되는 용어는 ABR이다. 그러나 CBR과 ABR은 과거의 경험을 활용하는 방법에서 차이가 있다. CBR이 단일 도메인을 대상으로

사례들에 대한 색인 및 대응전략에 초점을 둔 방법이라면, ABR은 다른 도메인의 과거 사례를 이용해서 새로운 문제를 해결하려는 방법이라는 점에서 구별된다. 따라서 ABR은 유사성 판단에 연구의 초점을 두고 있는 반면, CBR은 유용성에 연구의 초점을 두고 있다. 따라서 ABR에 관한 연구는 교차 도메인 상사들(cross-domain analogies)의 식별/활용을 위한 메커니즘의 하위분야라 할 수 있다. 또한 MBR은 기억장치의 조직화와 접근문제에만 연구의 초점을 두므로 주로 병렬처리기법과 관련이 된다. 그러나 CBR은 사례의 조직화 및 접근방법뿐만 아니라 사례의 활용방법에도 관련이 된다는 점에서 차이가 있다. 그의 EBR과 IBR은 적용분야에서 CBR과 차이가 있다. EBR은 개념정의를 위한 학습문제에 초점을 두는 것이며, IBR은 자동학습문제에 초점을 둔 것이다. 그러나 CBR은 학습뿐만 아니라 문제해결 등에도 적용가능하다.

#### IV. DSS와 사례기반 추론의 결합

##### 1. DSS와 사례기반 추론의 결합 필요성

DSS가 다른 정보시스템보다 효과적으로 의사결정지원을 할 수 있는 것은 다른 시스템과 달리 데이터베이스와 모형베이스를 이용하기 때문이다. 그동안 보다 효과적인 의사결정지원을 위해 데이터와 모형의 통합방법들에 대한 연구들이 진행되었다. Liang (1985)은 S/W 설계의 관점에서 모형관리와 자료관리의 통합 발전단계를 자료와 모형이 개별 응용프로그램으로 결합되는 1단계, 공통의 계산절차를 패키지로 통합하는 2단계, DB 관리언어에 의해 통합이 이루어지는 3단계, 그리고 모형관리시스템에 의해 통합이 이루어지는 4단계로 나누어 설명하였다. 즉, Liang은 통합관점에서 DSS에 있어서 보다 중요시 되어야 할 부분은 모형베이스 부분이라고 보았다. 이러한 추세에 따라 대부분의 DSS는 모형구축의 측면을 강조하기 때문에 Liang의 관점은 DSS 사용방식에도 잘 나타나 있다. 대부분의 의사결정자들은 경영과학에 대한 전문지식이 풍부하지 않다. 따라서 DSS 이용시 의사결정자들은 직접 시스템을 이용하지 않고 중개자(intermediary)를 통해 이용하는 것이 일반적인 사용방식으로 되어 있다(Turban, 1990).

그러나 중개자가 개입되는 DSS는 모형구축과 최적화만을 중요시 하기 때문에 모형의 구축 및 타당화 과정에 사용자의 참여를 제한함으로써 시스템의 이용, 해결안의 정확성

과 수용도를 저하시키게 된다. 이러한 문제는 모형구축의 어려움으로 인해 발생하는 것으로 시스템의 문제해결과정을 개선함으로써 해결될 수 있다. 즉, 기존의 DSS에서 사용자 대신 중개자가 수행하는 과업중 지식집약적이라 할 수 있는 모형구축과 최적해 생성 과정을 자동화할 경우, 시스템 사용에 따른 부담이 경감되어 사용자가 직접 시스템을 사용할 가능성이 높을 것이다.

AI 시스템에서는 모형수립과 해결안 생성이 시스템에 의해 수행된다. 따라서 DSS 이용시 중개자에 대한 의존성을 줄여주기 위해서는 AI 기술이 도입되어야 하며, 이러한 시스템은 보다 효과적인 문제해결을 가능하게 한다(Elam & Konsynski, 1987). 이는 의사결정과정에 DSS와 AI 기술이 결합된 새로운 형태의 DSS, 즉 지적 DSS의 도입 필요성을 나타내는 것이다. 따라서 기존의 DSS들이 문제해결의 전단계를 지원하지 못하고 정찰과 선택의 단계만을 지원함에 따라 DSS의 고유 목적인 문제해결의 전단계를 지원 가능하게 하기 위해서는 AI 기술과의 결합이 필요함을 의미하는 것이다. CBR 기법은 사례를 통해 문제를 해결하려는 방법이다. 사례는 문제해결에 필요한 데이터와 모형을 문제기술부분과 해결안부분으로 가지고 있다. CBR 기법은 부분일치법(partial matching)을 이용하므로 문제에 대한 부분적인 자료만을 가지고도 관련 사례를 검색할 수 있으며, 검색된 사례들중에서 기저사례를 선택하므로 문제해결에 필요한 데이터의 수집과 모형수립문제는 자동적으로 지원되게 된다. 따라서 CBR 기법은 의사결정과정의 정찰과 선택 단계를 지원해 줄 수 있다. 또한 각 사례는 하나의 의사결정대안이 될 수 있다. 사례검색과정은 문제해결에 유용한 다수의 관련 사례들을 제시해주므로, 이는 대안의 생성과도 같다. 따라서 사례검색과정을 통해 의사결정과정의 설계단계도 지원이 가능하다.

## 2. 의사결정지원 도구로서의 사례기반 추론

DSS는 비구조적 혹은 반구조적 의사결정문제를 지원하기 위한 도구이다. 따라서 DSS는 AI 시스템과 달리 의사결정자를 대신하는 것이 아니라 의사결정자를 지원해주는 시스템이다. 기존의 DSS에 관한 연구들은 의사결정단계중 상대적으로 정찰과 선택의 단계에 중점을 두었다. 따라서 기존의 연구들은 주로 모형구축과 자료분석을 중요시함에 따라 계량화할 수 있는 문제에 대해서만 연구의 초점을 두었으며, 계량화할 수 없는 문제에 대한 연구는 등한시 하였다. 이로 인해 CBR 기법과 같은 일반적인 문제해결방법은

DSS의 연구분야에서 무시되고 있었다.

CBR 기법은 AI분야의 문제해결기법중 하나이다. 현재까지 CBR 연구의 초점은 문제 해결 자체에 있었지 문제해결의 지원은 고려되지 않았다. 그 이유는 CBR 기법이 AI기법에 속하기 때문에 CBR들도 AI시스템이 취하는 일반적 문제해결방식에 따라 사용자를 시스템이 주도해 나가도록 설계되었기 때문이다. 그러나 이러한 전통적인 관점에서 벗어나 Kolodner(1991)는 사람과 기계가 함께 문제를 해결하기 위한 시스템 구축방법론으로 사례기반형 의사결정지원(case-based decision aiding)이라는 개념을 제시하였다. 사례기반형 의사결정지원방식에서는 문제해결에 이용될 사례를 제공함으로써 사람의 기억기능을 돕는다. 사람은 제공된 사례들을 지표(guidelines)로 삼아 문제해결을 위한 의사결정을 수행한다. 만약 최종 의사결정은 사용자가 하고, 올바른 사례의 검색은 시스템이 지원할 경우 사용자의 참여없이 단순히 최종결과만을 제시하는 전통적 CBR보다는 사례기반형 의사결정지원방식에 의한 시스템에 대한 채택 및 의존율이 높을 것이다.

현재 다수의 CBR들이 개발되어 법률, 진단, 분쟁중재, 요리, S/W 비용추정 등 다양한 분야에 응용되고 있다. 그러나 이러한 CBR들은 대부분 CBR 기법이라는 하나의 관점에서 문제해결이나 상황의 해석 등을 다루었다. 따라서 그동안 DSS의 관점과 CBR의 관점을 통합하려는 시도는 거의 없었다. 최근 Sinha 등(1996)이 Kolodner가 제시한 의사결정지원의 관점에서 CBR 기법을 이용한 연구를 하였다. 이들은 일상적 설계(routine design)문제를 대상으로 제약식전파(constraint posting)기법과 CBR 기법을 결합해서 IDEA라는 시스템을 개발하였다. IDEA에서는 먼저 문제가 입력되면 이를 원시문제(primitives)로 분해한 후, 각 원시문제에 대한 제약식을 부여해서 탐색공간을 제한하게 된다. 문제가 분해되고 탐색공간이 결정되면 CBR 기법을 이용해서 관련 사례들을 검색한 후, 검색된 사례들을 효용함수를 통해 평가함으로써 최적의 사례를 선택한다. Sinha 등의 연구는 응용도메인이 일상적인 설계라는 구조적 문제를 대상으로 하였기 때문에 DSS의 주요 지원영역인 반구조적 혹은 비구조적 문제의 지원을 고려하지 않았다. 그러나 CBR 기법이 효과적으로 의사결정을 지원하기 위해서는 구조적 문제보다는 반구조적 혹은 비구조적 문제해결의 지원이 가능하도록 DSS와 결합되어야 할 것이다.

### 3. 사례기반 DSS의 구조

시스템 구조(system architecture)란 S/W의 각 구성요소들을 연결한 형태를 의미하는 것으로 DSS의 경우에는 연구의 초기단계인 '70년대 초에 많은 형태들이 제시되었다. 현재 DSS의 구조는 대체로 인터페이스, 데이터베이스, 모형베이스라는 3가지 구성요소를 갖는 것으로 받아 들여지고 있다. 그러나 현재 CBRS에 대한 연구는 초기단계이므로 시스템 구조에 대한 통일된 견해는 없으며 시스템에 따라 다양한 형태를 나타내고 있다. 예를 들면, Simoudis(1992)는 DEC사의 VMS 운영체제에 영향을 미치는 장치드라이버의 실패문제를 해결하기 위한 CBRS인 Cascade의 구성요소로 사용자 인터페이스, 표면속성 검색기, 검색된 사례들로부터 추가적인 문제특성을 알아내기 위한 모형기반형 타당성평가기, 표면지식베이스, 타당성평가모형, CB를 이용하였다. Yoon 등(1993)은 방사능치료기의 수리를 위한 파견기술자 선정을 위한 CBRS인 SERVICE의 구성요소로 사례베이스 관리기, 서비스요청 관리기, 사례검색기, 고객 DB, CB를 이용하였다. McCarthy(1993)는 식사준비를 위한 CBRS인 COOKIE의 구성요소로 계획생성기 모듈, 실행감시기 모듈, 계획수정기 모듈, 색인부여 모듈, CB를 이용하였다. Gupta(1994)는 CBRS의 구성요소로 CB, 색인 라이브러리, 관련성측도모듈, 설명모듈을 제안하였다. Mahapatra 등(1994)은 CBRS의 일반적인 구성요소로 사례추론기, 유지보수기, CBMS, CB를 제안하였다. Xu (1995)는 AIDS의 초기진단을 위한 CBRS의 구성요소로 입력모듈, 사례기억장치, 사례검색기를 이용하였다. Hinkle 등(1995)은 압력용기의 처리율을 높이기 위한 CBRS인 CLAVIER의 구성요소로 인터페이스, 검색기, CB, 그래픽에디터 및 검정기, DB와 스케줄관리를 위한 계획수립 및 스케줄러를 이용하였다. Paek 등(1996)은 DB 스키마 설계를 위한 CBRS인 DES-DS의 구성요소로 인터페이스, RCG 번역기, 사례검색기, 스키마 설계기, CB를 이용하였다. 특히 이들은 CB를 도메인 독립적인 것과 종속적인 것으로 구분하였다.

이러한 시스템 구조의 다양성은 응용도메인이 각각 다르기 때문에 발생할 수 있다. 그러나 시스템에 따라 명칭은 다르나 대부분의 구성요소들이 유사한 기능들을 수행하고 있다. 따라서 이들 구성요소들을 기능별로 분류해 보면 <표 4>와 같다. <표 4>에서 볼 수 있듯이 구성요소들의 명칭은 시스템에 따라 차이가 있으나 공통적으로 언급되는 것은 사용자 인터페이스, 사례추론기, CB 등이다. 따라서 CBDSS가 효과적으로 의사결정을 지원하기 위해서는 이들 공통적인 구성요소들과 사례추론기가 활용하는 관련 지식베

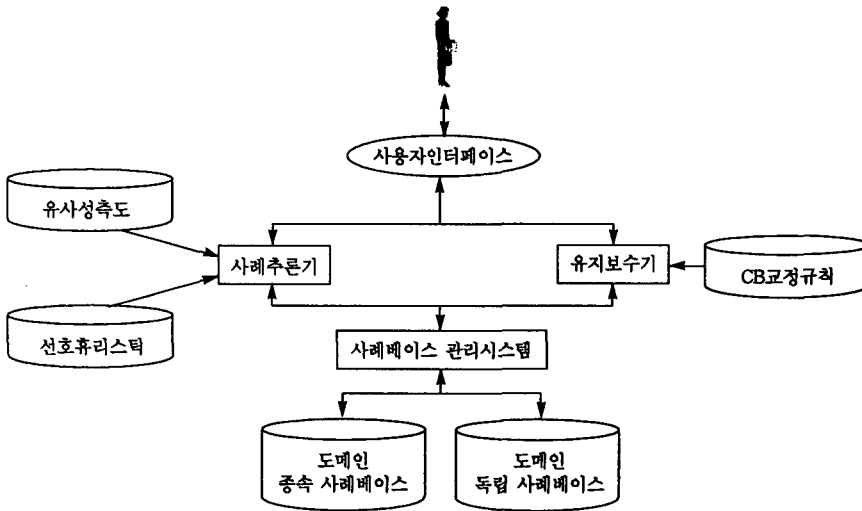
〈표 4〉 CBRs의 구성요소들에 대한 기능별 분류

연구자	시스템명	인터페이스	사례추론기	CB	기타
Simoudis	Cascade	· 사용자 인터페이스	· 표면속성검색기 · 모형기반형 타당성평가기	· CB	· 표면지식베이스 · 타당성평가모형
Yoon 등	SERVICE		· 사례검색기	· CB	· 서비스요청 관리기 · 고객 DB
McCarthy	COOKIE		· 계획생성기모듈	· CB	· 색인부여모듈 · 실행감시기모듈 · 계획수정기모듈
Gupta	-		· 관련성측도 모듈	· 사례지식 베이스	· 색인라이브러리 · 설명모듈
Mahapatra 등	-		· 사례추론기	· CB	· 유지보수기 · CBMS
Xu	-	· 입력모듈	· 사례검색기	· 사례기억장치	
Hinkle 등	CLAVIER	· 그래픽사용자 인터페이스	· 검색기 · 계획수립 및 스케줄러	· CB	· 그래픽에디터 및 점정기 · DB
Paek 등	DES-DS	· 인터페이스	· 사례검색기	· DDCB · DICB	· RCG 번역기 · 스키마설계기

이들이 필요할 것이다. 또한 효율적인 CB의 관리를 위해서는 Mahapatra 등이 제시한 사례베이스관리시스템과 유지보수기, Paek 등이 제안한 CB의 분리방법이 채택될 필요가 있다. 따라서 CBDSS의 구조는 기존의 DSS와 매우 상이한 구조를 가지게 된다. 본 논문에서 제시하는 CBDSS의 구조는 〈그림 5〉와 같다.

Turban 등(1986)은 ES와 DSS의 통합문제에 대해 개별 기능지원을 위해 ES를 DSS의 개별 구성요소와 결합하는 통합 프레임워크와 ES를 DSS의 독립적인 구성요소로 결합하는 통합 프레임워크를 제시하였다. 제안한 CBDSS의 구조는 Turban 등이 제시한 ES와 DSS의 통합 프레임워크에서 볼 때 ES를 DSS의 개별 구성요소의 기능보강을 위해 이용한 경우이다. 즉, 제안한 CBDSS에서는 CBR 기법을 이용해서 데이터베이스와 모형베이스 및 사용자를 지원할 수 있도록 하였다. 그 결과 데이터베이스와 모형베이스는 CB로 대체되





〈그림 5〉 CBDSS의 구조

있으며, 이들을 관리하는 데이터베이스관리시스템(Database Management System : DBMS)과 모형베이스관리시스템(Modelbase Management System : MBMS)은 사례베이스관리시스템(Casebase Management System : CBMS)으로 대체되었다. 또한 사용자의 선호도를 반영하고 시스템의 문제해결능력을 향상시키기 위해 유사성 측도, 선호휴리스틱, 추상화 계층구조 등의 지식베이스를 이용하는 사례추론기가 도입되었다.

#### 4. 사례기반 DSS의 구성요소

##### 4.1 CB와 관련요소

CBDSS에서는 CBR 기법을 이용해서 의사결정을 지원하므로 CB를 구성요소로 갖는다. CB의 각 사례는 문제내용과 문제해결에 이용할 전략부분으로 구성되어 있다(Hurley, 1994). 즉, 문제의 특성과 해결방법이 하나의 사례에 통합되어 표현된다. 따라서 CBDSS에서는 전통적인 DSS에서와 같이 문제의 특성을 나타내는 자료부분과 해결방법에 해당하는 모형부분을 데이터베이스와 모형베이스에 분리 저장할 필요가 없다.

전통적인 CBR 기법에서는 응용도메인에 대한 경험적 지식만을 이용해서 문제를 해결하려 하였다. 그러나 이러한 전통적인 사고에서 벗어나 Gupta(1994)와 Paek 등(1996)은 일반지식을 문제해결에 이용하려고 하였다. Gupta는 문제해결에 필요한 경험과 원칙들

을 단일의 사례지식베이스(case knowledge base)에 저장해두고 관련성측도 모듈을 이용해서 문제해결에 이용하려 하였다. Paek 등은 DB 스키마설계에 대한 경험적 지식은 도메인 종속 CB(Domain Dependent CB : DDCB)에 저장하고, 정규화 지식(normalization knowledge)은 도메인 독립 CB(Domain Independent CB : DICB)에 분리 저장한 후 문제해결에 이용하려 하였다. CBR 기법에 일반지식을 이용함으로써 문제해결능력을 향상시키려는 의도는 Gupta와 Paek 등이 모두 동일하나 사례의 관리와 검색의 효율성 측면에서 보면 Paek 등에 의한 CB의 분리방식이 보다 우수하다고 할 수 있다. 따라서 CBDSS의 경우에도 의사결정지원 능력을 향상시켜 주기 위해 과거의 의사결정지원 경험들은 DDCB에 저장하고 의사결정지원을 위한 일반적인 지식들은 DICB에 분리해서 저장하는 것이 효율적일 것이다.

일반지식을 규칙베이스로 구축하지 않고 CB 형태로 구축하는 이유는 추론메커니즘을 추가로 도입하지 않기 위해서이다. 만약 일반지식을 규칙베이스에 저장할 경우 이들의 표현 형식은 생성규칙이 될 것이므로, 관련 지식을 검색하기 위해서는 RBS에서 이용되는 추론메커니즘이 추가적으로 필요하게 된다. 따라서 일반지식을 사례형식으로 표현하여 DICB를 도입하면 DDCB에서와 동일한 추론방식을 이용할 수 있으므로 구축의 측면에서 볼 때, 규칙베이스를 이용하는 것보다 합리적이라 할 수 있다.

CB는 전통적 DSS의 구성요소중 데이터베이스와 모형베이스에 대응되는 부분이다. DB가 데이터를 보관하고 있는 곳이라면 CB는 사례를 보관하고 있는 곳이다. 사례도 데이터와 같이 조직의 자원이다. 그러므로 다수의 사용자에게 의해 사례들이 이용될 수 있어야 한다. 그리고 사례는 데이터보다 구조적으로 복잡한 형태이므로 사례의 효율적 관리는 중요한 문제가 된다. 또한 CB에는 시간이 경과함에 따라 많은 사례들이 누적된다. 즉, 사례의 수가 증가되어 CB의 복잡성이 증가하게 된다. 따라서 이들을 효율적으로 관리하기 위한 도구로서 CBMS가 필요하게 된다. CBMS는 응용영역 독립적인 구성요소로서, 이의 기능은 전통적 DSS 아키텍처에 있어서 데이터베이스 모듈의 DBMS와 모형베이스 모듈의 MBMS의 기능을 대신하는 부분이다. CBMS는 사례추론기와 유지보수기의 지시에 따라 관련 사례들을 DDCB와 DICB에서 추출하는 역할과 CB의 관리를 담당하는 역할을 수행한다. 그러나 CBMS에서는 기존의 DSS에서 통합의 단계에 따라 DBMS 혹은 MBMS가 담당하던 관련 데이터와 모형의 결합을 위한 기능은 필요가 없다. 왜냐하면 CBDSS에서는 의사결정지원에 필요한 모든 정보가 사례라는 단일 구조에 포함되어 있기

때문이다. 따라서 CBMS의 주요 기능으로는 CB의 사례들을 추가, 삭제, 갱신, 검색 등이 있다.

#### 4.2 사례추론기와 관련요소

CBDSS에 있어서 사례추론기(Case-based Reasoner : CBRer)는 시스템 전체의 성능에 가장 많은 영향을 미치는 구성요소이다. CBRer는 지적 DSS에서 가장 중심적인 역할을 수행하는 문제처리에 해당한다. CBRer의 역할은 CB에 있는 사례들중 현문제와 가장 유사한 사례를 선별해주는 것이다. 시스템에 따라 CBRer가 수행하는 기능에는 차이가 있을 뿐만 아니라 구현에 있어서도 단일 구성요소로 혹은 기능별로 다수의 구성요소로 분리되어 있다. CLAVIER 시스템의 경우 CBRer는 검색기와 계획수립 및 스케줄러로 구성되어 있다. 이 시스템에서 검색기가 수행하는 역할은 단순히 CB에서 관련 사례를 추출하는 것이며, 추출된 사례를 문제해결에 이용하는 과정은 계획수립 및 스케줄러가 담당하도록 되어 있다.

Cascade 시스템의 경우 CBRer는 표면속성 검색기와 모형기반형 타당성평가기로 분리되어 있다. 표면속성 검색기는 표면지식베이스를 이용해서 CB에서 관련 사례들을 추출한 후 이를 모형기반형 타당성평가기에 전달하게 된다. 1차 검색된 사례들을 대상으로 모형기반형 타당성평가기는 문제에 대한 추가적인 속성을 파악한 후 CB를 재검색하게 되며 재검색된 사례들중에서 기저사례를 선택하게 된다. 선택된 기저사례는 사용자와의 상호작용을 통해 해결안 수립에 이용된다.

Xu가 개발한 시스템의 경우 CBRer는 색인/일치모듈(indexing/matching module), 근사선택모듈(approximate selection module), 수정모듈(adaptation module)로 구성되어 있다. 색인/일치모듈은 주요 속성을 이용해서 관계형 DB형식으로 되어 있는 CB에서 관련된 사례들을 추출하게 된다. 추출된 사례들은 근사선택모듈에 입력되어 퍼지평가함수와 사용자와의 상호작용을 통해 수정이 용이한 사례가 기저사례로 선택된다. 선택된 기저사례는 수정모듈에 입력되어 수정규칙과 사용자의 도움을 받아 새로운 문제에 적합한 해결안으로 수정된다.

제한한 CBDSS에서는 사용자 인터페이스를 통해 새로운 문제가 입력되면 CBRer가 먼저 관련된 사례들을 식별하기 위해 CBMS의 도움을 받아 문제에 적합한 사례를 DDCB

혹은 DICB에서 추출하게 된다. 추출된 사례들중 기저사례를 선택하는 과정은 최종해결안의 질에 많은 영향을 미치게 된다. 따라서 기저사례는 문제해결에 유용한 사례가 되어야 한다. 일반적으로 기저사례의 선택에는 수치적 평가함수가 이용되나 이는 높은 평가를 받은 사례가 그렇지 못한 사례에 비해 우수한 이유를 설명하지 못한다. 따라서 CBDSS의 CBRer는 문제해결에 유용성이 높고 사용자의 선호를 충족시켜 줄 수 있는 기저사례의 선택을 위해 선호휴리스틱절차를 이용하는 것이 유리할 것이다. 따라서 CBDSS에서는 CBRer가 다양한 기능을 가진 단일의 구성요소가 되며, 문제해결을 위해 관련 지식베이스를 이용하게 된다.

#### 4.3 유지보수기와 기타 구성요소

CBR은 문제해결시 과거 경험을 이용함으로써 문제해결에 소요되는 시간과 비용을 절약할 수 있다. 그러나 CBR의 단점중 하나는 저장된 사례가 부정확한 사례일 경우 이를 반복적으로 사용함으로써 동일한 실수를 반복하게 된다는 것이다. 따라서 이러한 문제를 해결하기 위해서는 도메인 전문가에 의한 해결안의 검정과 타당성 평가가 있어야 한다.

또한 CB는 동적 진화의 특성이 있다. 따라서 시간이 경과함에 따라 CB에 새로운 사례가 추가되거나 기존의 사례가 삭제 혹은 수정되게 된다. CB내의 사례들은 상호 연결된 형태로 저장되어 있다. 기존의 사례가 삭제됨으로써 삭제된 사례와 연결된 하위단계의 사례들은 다른 사례들과 분리되는 현상이 발생할 수 있다. 만약 이러한 문제가 발생할 경우에는 문제해결에 유용한 사례가 CB에 존재해도 검색을 하지 못할 수 있다. 이러한 현상을 막아주기 위해서는 CB의 진화과정을 지원하기 위해 유지보수기가 필요하게 된다. 유지보수기는 도메인 전문가가 주로 사용하는 부분으로 사례의 타당성 평가 및 사례의 추가, 삭제, 변경에 따른 사례의 고립(isolated) 문제 등을 해결해 주는 하위시스템으로서 CB 교정규칙을 이용한다. 현재 개발된 대부분의 시스템에서는 아직 유지보수기에 대한 부분이 고려되지 않았다. 그러나 CBR 기법을 이용해서 효과적으로 의사결정을 지원하기 위해서는 유지보수기도 하나의 구성요소로 도입될 필요성이 있다.

사용자 인터페이스는 사용자가 해결할 문제를 시스템에 입력하고 시스템의 실행결과를 사용자에게 제시하는 곳으로 시스템과 사용자간의 정보교환을 가능하게 하는 부분이다. 초기의 사용자 인터페이스는 주로 문자 형식으로 정보를 표현하였으나 최근에는 사

용자의 편의성을 위해 그래픽 방식으로 정보를 표현하는 것이 일반화되었다. 사용자 인터페이스와 연결되는 하위시스템에는 일반 사용자가 주로 이용하는 CBRer와 도메인 전문가가 주로 이용하는 유지보수기가 있다.

제안한 CBDSS 구조에서는 CBRs의 주요 구성요소중 수정규칙, 색인규칙 등에 대한 지식베이스가 고려되지 않았다. 수정규칙을 배제시킨 이유는 초기 CBRs는 자율적 문제 해결시스템(autonomous problem solver)이었으나 최근의 CBRs들은 사용자를 위한 상호작용식 외부 기억장소(interactive external memory)로 이용되기 때문이다(Kolodner, 1992). 즉, 상호작용방식으로 인해 과거 해결안을 수정하는 일이 CBRer로부터 사용자에게로 이전되게 되었기 때문에 수정규칙이 시스템의 구성요소로 존재할 필요가 없다. 또한 색인규칙을 통한 사전 색인부여방식(preindexing)은 사례검색의 유연성과 CB의 일반적 활용을 제한한다. 그리고 심리학적 연구 측면에서 볼 때 인간은 여러 가지 유사성 개념의 특성을 결합해서 상사검색을 하기 때문에 특정 측면에 의한 색인부여의 기대효과가 의문시되고 있다(Kettler, Hendler, Anderson & Evett, 1994). 따라서 이러한 색인부여방식의 문제점에 따라 소형 CB의 경우에는 색인부여를 않는 것이 사례활용도를 증대시켜 줄 것이며, 대형 CB의 경우에는 병렬처리방식을 이용하는 것이 일반적이다.

## V. 결 론

CBR은 과거의 경험을 통해 문제를 해결하려는 기법으로서 규칙기반 추론과는 달리 의사결정 시점마다 반복적으로 해를 도출하지 않는다. 따라서 CBR은 규칙기반 추론보다 상대적으로 문제를 신속히 해결해 줄 수 있다. 또한 CBR은 당면 문제가 과거의 경험과 정확히 일치하지 않더라도 부분적인 해결안을 제시해 줄 수 있는 기법이다. DSS는 의사결정에 필요한 자료와 모형을 자료베이스와 모형베이스에 저장해두고 의사결정문제가 발생할 때마다 필요한 자료와 모형을 결합해서 새로운 해결안을 제시해주는 시스템이다. 즉, DSS는 규칙기반 추론과 같이 동일한 문제에 대해서도 의사결정시점마다 반복적으로 해를 도출해야 한다.

따라서 CBR과 DSS의 결합시스템인 CBDSS에서는 신속한 문제해결을 위해 CBR의 문제해결방식을 채택하는 것이 효율적일 것이다. CBR에서는 문제해결에 필요한 자료와 모형을 사례형식으로 표현해서 사례저장소에 저장한다. 그러므로 CBDSS에서는 CBR의 문

제해결방식을 이용함으로써 전형적인 DSS의 구조에서, 독립적으로 존재하던 자료와 모형이 사례로 통합되어 존재하게 되며, CB의 관리를 위해 MBMS와 DBMS 대신 CBMS가 도입된다. CBDSS에는 지적 DSS의 문제처리기에 해당되는 사례추론기가 도입되며, 관련 사례들을 효율적으로 활용하기 위해 사례추론기가 이용하는 관련 지식베이스들이 도입되어야 한다. 유지보수기는 CB의 동적 진화를 지원해주기 위해 필요한 구성요소로서 이의 존재로 인해 CB의 효율적 관리기능이 보다 향상될 수 있을 것이다.

## 參 考 文 獻

1. Aamodt, A., and E. Plaza, "Case Based Reasoning : Foundational Issues, Methodological Variations, and System Approaches," *Artificial Intelligence Communications*, Vol. 7, No.1, 1994, pp.39-59.
2. Alter, S. A., "A Taxonomy of Decision Support Systems," *Sloan Management Review*, Vol.19, No.1, 1977, pp.39-56.
3. Alterman, R., "Adaptive planning," *Cognitive Science*, Vol.12, 1988, pp.393-421.
4. Alterman, R., "Panel Discussion on 'Case Representation'," *Proceedings : Case-Based Workshop*, May, 1989, p.15.
5. Elam, J. J., and B. Konsynski, "Using Artificial Intelligence Techniques to Enhance the Capabilities of Model Management Systems," *Decision Science*, Vol.18. 1987. pp.487-501.
6. Fedorowicz, J., and G. B. Williams, "Representing Modeling Knowledge in an Intelligent Decision Support Systems," *Decision Support Systems*, Vol.2, 1986, pp.3-14.
7. Gupta, U. G., "How Case-Based Reasoning Solves New Problems," *Interfaces*, Vol.24, No.6, 1994, pp.110-119.
8. Hinkles, D., and C. Toomey, "Applying Case-Based Reasoning to Manufacturing," *AI Magazine*, Vol.16, No.1, 1995, pp.65-73.
9. Hurley, N., "A case based reasoning approach to mesh specification for adaptive finite element analysis," *Mathematics and Computers in Simulation*, Vol.36, 1994, pp.381-388.
10. Kettler, B. P., J. A. Hendler, W. A. Anderson, and M. P. Evett, "Massively Parallel Support for Case-Based Planning," *IEEE Expert*, February, 1994, pp.8-14.
11. Kolodner, J., "Improving Human Decision Making through Case-Based Decision Aiding," *AI Magazine*, Summer, 1991, pp.52-68.
12. Kolodner, J., *Case-Based Reasoning*, Morgan Kaufmann, San Mateo, CA, 1993.
13. Kolodner, J., and W. Mark, "Case-Based Reasoning," *IEEE Expert*, 1992, pp.5-6.
14. Liang, T., "Integrating Model Management with Data Management in Decision Support Systems," *Decision Support Systems*, Vol.1, 1985, pp.221-232.

15. Mahapatra, R., and A. Sen, "Case Base Management Systems : Providing Database Support to Case-based Reasoners," *J. of Database Management*, Vol.5, No.2, 1994, pp. 19-29.
16. McCarthy, R., "Episodic Cases and Real-Time Performance in a Case-Based Planning System," *Expert Systems With Applications*, Vol.6, 1993, pp.9-22.
17. McGovern, J., D. Samson, and A. Wirth, "Using case based reasoning for basis development in intelligent decision systems," *European J. of Operational Research*, Vol.74, 1994, pp.40-59.
18. Mott, S., "Case-Based Reasoning : Market, Applications, and Fit With Other Technologies," *Expert Systems with Applications*, Vol.6, 1993, pp.97-104.
19. Paek, Y., J. Seo, and G. Kim, "An expert system with case-based reasoning for database schema design," *Decision Support Systems*, Vol.18, 1996, pp.83-95.
20. Simoudis, E., "Using Case-Based Retrieval for Customer Technical Support," *IEEE Expert*, October, 1992, pp.7-12.
21. Sinha, A. P., and J. H. May, "Providing Design Assistance : A Case-Based Approach," *Information Systems Research*, Vol.7, No.3, 1996, pp.363-387.
22. Slade, S., "Case-Based Reasoning : A Research Paradigm," *AI Magazine*, Spring, 1991, pp.42-52.
23. Sprague, R. H., "DSS in Context," *Decision Support Systems*, Vol.3, 1987, pp.197-202.
24. Turban, E., *Decision Support and Expert Systems : Management Support Systems*, 2nd ed., Macmillan Publishing Company, N. Y., 1990.
25. Turban, E., and P. R. Watkins, "Integrating Expert Systems and Decision Support Systems," *MIS Quarterly*, June, 1986, pp.121-136.
26. Xu, L. D., "Case-based reasoning for AIDS initial assessment," *Knowledge-Based Systems*, Vol.8, No.1, 1995, pp.32-38.
27. Yoon, Y., A. D. Acree, and L. L. Peterson, "Development of a Case-Based Expert System : Application to a Service Coordination Problem," *Expert Systems with Applications*, Vol.6, 1993, pp.77-85.



## Abstract

## Integrating Case-Based Reasoning with DSS

Kim, Jin-baek

Case-based reasoning(CBR) offers a new approach for developing knowledge based systems. Unlike the rule-based paradigm, in which domain knowledge is encoded in the form of production rules, in the case-based approach the problem solving experience of the domain expert is encoded in the form of cases stored in a casebase(CB).

CBR allows a reasoner (1) to propose solutions in domains that are not completely understood by the reasoner, (2) to evaluate solutions when no algorithmic method is available for evaluation, and (3) to interpret open-ended and ill-defined concepts. CBR also helps reasoner (4) take actions to avoid repeating past mistakes, and (5) focus its reasoning on important parts of a problem. Owing to the above advantages, CBR has successfully been applied to many kinds of problems such as design, planning, diagnosis and instruction.

In this paper, I propose case-based DSS(CBDSS). CBDSS is an intelligent DSS using CBR technique. CBDSS consists of interface, case-based reasoner, maintainer, casebase management system, domain dependent CB, domain independent CB, and so on.