
 論 文

大韓造船學會論文集
 第29卷 第1號 1992年 3月
 Transactions of the Society of
 Naval Architects of Korea
 Vol. 29, No. 1, March 1992

선체 구조설계를 위한 지식 베이스 전문가 시스템

양영순*, 연운석**

Knowledge-based Expert System for the Preliminary Ship Structural Design by

Y. S. Yang* and Y. S. Yeon**

요 약

본 논문은 선박의 초기설계와 중앙단면 설계용 지식베이스 시스템을 개발하고 지식공학이 선체구조 설계 분야에 매우 효율적으로 적용될 수 있음을 검토하는데 그 목적이 있다.

이를 위하여 첫째 구조설계에 적합한 지식베이스 전문가 시스템 개발용 도구(tool)인 E.I를 개발하였다. E.I는 C언어로 작성되었고 구조설계 과정을 효율적으로 뒷받침해주기 위해서 자동 반복 계산과 재추론 기능을 갖고 있다. 둘째 살물선의 초기설계시 주요치수 추정과 중앙단면 설계를 위한 지식베이스를 개발하였다. 이 지식베이스들은 통계적 경험식, 선급 규정, 경험적 지식 등을 rule, fact, table로 표현하여 구성하였다. 셋째 지식베이스의 추론을 통하여 중앙 단면 최적 설계 프로그램의 입력 자료를 생성하고 최적화 프로그램을 수행시키는 통합(hybrid) 시스템을 구축하였다. 넷째 보강판 설계용 지식베이스 시스템을 개발하기 위해 최적화 결과의 회귀분석을 이용하는 간이 설계법을 적용해 보았다.

이 시스템을 사용하여 살물선과 보강판의 설계과정을 보이고 그 결과를 검토해 보았다. 앞으로 E.I의 성능 개선과 더 우수한 지식베이스가 축척됨에 따라서 선체 구조설계에 사용되는 기존의 CAD 시스템의 성능이 크게 보완될 것으로 기대된다.

Abstract

The objective of this study is to develop knowledge-based system for the preliminary design and midship section design of bulk carrier and to enhance the applicability of knowledge engineering in the field of

발표: 1991년도 대한조선학회 춘계연구발표회('91. 4. 13.)

접수일자: 1991년 5월 20일, 재접수일자: 1991년 7월 22일

*정회원, 서울대학교 조선해양공학과

**학생회원, 서울대학교 조선해양공학과

Naval Architecture.

First, expert system shell called E.1 is developed in C language. E.1 supports backward-chaining, automatic iteration procedure and reiterative inference mechanism for efficient application of knowledge-based system in structural design. Knowledge representation in E.1 includes IF-THEN rules, 'facts' and 'tables'. Second, knowledge bases for the principal particulars and midship section design are developed by experimental formula, design standard and experiential knowledge. Third, hybrid system combined this knowledge-based system with the optimization program of midship section is developed. Finally, the simplified design method utilizing the regression analysis of the optimum results of stiffened plate is developed for facilitating the design process.

Using this knowledge-based system, the design process and results for Bulk carrier and stiffened plates are discussed. It is concluded that knowledge-based system is efficient for preliminary design and midship section design of the ship. It is expected that the performance of the CAD system would be enhanced if the better knowledge-base is accumulated in the E.1 tool.

1. 서 론

최근 설계업무에 사용되는 CAD시스템에 인공지능(artificial intelligence)을 사용함으로써 창조적이고 지적인 기능을 포함하려는 시도가 많이 진행되어 현재 대화식 추론 시스템(interactive deduction inference system)이 활용되고 있다[1] [2] [3] [4] [5]. 이러한 추론 시스템의 적절한 적용은 기존 시스템의 성능을 향상시키거나, 시스템의 유지, 관리, 지속적인 개발에 수반되는 여러 문제점을 해결해 줄 수 있어 최근 크게 관심이 모아지고 있다. 본 연구에서는 지식베이스 시스템을 선박의 초기설계와 중앙단면 설계 그리고 보강판 설계에 적용하여 선체구조설계에 인공지능 기법의 이용 가능성을 검토해 보았다[6] [7]. 선박의 초기설계 과정에 필요한 데이터들은 대부분 불확실한 값들이며 이 데이터를 준비하기 위해서는 경험적 지식을 필요로 한다. 또 주요치수 결정을 위한 식들이 과거의 설계 경험에 근거한 통계적인 경험식들이기 때문에 사용된 경험식들에 따라서는 같은 데이터를 사용하더라도 다른 결과를 줄 수 있다. 더우기 설계 지식과 경험이 축적됨에 따라서 경험식들은 계속해서 수정되는 경향을 갖는다. 특히, 중앙단면의 기본 구조를 결정하는 단계는 전산화하기가 힘든 과정이기 때문에 대부분 설계자 개

인의 지식과 유사선의 실적 자료에 의해 수작업으로 이루어지고 있다. 중앙단면 설계는 주로 선급 규정을 사용하여 이루어지는데 선급 규정에는 불분명하고 체계화되지 않은(i11-defined) 요소가 많아 설계 초보자가 설계를 할 경우 많은 애로점을 느끼게 되어, 이를 극복하기 위한 방법으로써 인공지능 기법의 채용이 적극적으로 모색되고 있다. 그래서 살물선(bulk carrier)의 초기설계 및 중앙단면 설계용 지식베이스 시스템을 개발하였고 이를 통하여 지식 공학이 선체 구조설계에 충분히 적용될 수 있음을 검토해 보았다.

구조물의 설계시 보통 최적화 방법이 많이 사용되는데 최적화 방법은 복잡한 해석 과정을 수반하므로 계산시간이 많이 소요되는 것이 일반적이다. 그래서 설계시 많은 수치계산 과정을 피하고 최적해에 가까운 설계 결과를 얻을 수 있는 간이 설계법의 채용이 절실하다. 본 논문에서는 지식베이스 시스템을 토대로 한 보강판의 설계를 통하여 간이 설계법의 타당성을 검토해 보았다.

2. 지식베이스 시스템 개발용 도구(E.1)

2.1. E.1의 구성 요소

E.1의 구성 요소는 추론 기구, 지식베이스 loader, 설명기구, 사용자 인터페이스 등으로 구분되며

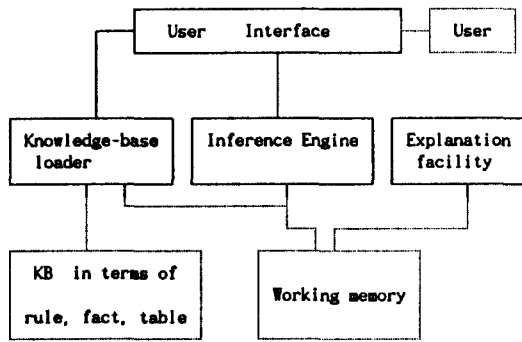


Fig. 1 Basic structure of E.1

이것을 도식화하면 Fig. 1과 같다.

E.1의 추론 기구는 주어진 문제 영역을 후향 추론(backward chaining)에 의해서 탐색해 나간다. 즉 pattern-matching에 의한 탐색(search)기능을 이용하여 문제의 해를 얻게 된다.

지식베이스 loader는 지식베이스를 추론 기구가 추론할 수 있는 형식으로 변환하여 컴퓨터의 기억 장소에 저장하며 Working memory 영역을 확보한다. 이때 지식베이스가 규정된 구문(syntax)에 위배되는 지를 검사한다. 시스템의 빠른 수행 속도는 수치계산이 많은 공학의 응용 분야에서는 필수적 요구 사항이다. 이것을 위하여 지식베이스 loader는 지식 베이스의 infix operator를 postfix 형태로 변환 시키고 변수들에 대한 symbol table을 만들어 추론 기구의 추론 효율을 크게 향상 시키도록 하였다. 지식베이스의 loader에 사용된 parser는 picture parsing기법을 사용하여 설계하였다 [8] [9] [10] [11]. 지식베이스의 구문이 비교적 단순하기 때문에 picture parsing기법은 충분히 효율적이다.

설명 기구는 얻어진 결과나 추론과정을 사용자가 요구시 설명하는 역할을 한다. 일반적으로 지식베이스 시스템에서는 3가지 기능 즉 'what', 'how', 'why' 등으로 분류하고 있지만 E.1의 현재 version은 아직 부분적인 'why' 기능만을 갖고 있다.

지식베이스는 살물선의 주요치수 결정과 중앙 단면의 각 부재 치수를 결정하고 중앙단면 최적화

프로그램의 입력 file을 만드는 rule, fact, table로 이루어져 있으며 E.1의 수행시 컴퓨터의 기억 장소로 지식베이스 loader에 의해 저장된다.

사용자 인터페이스는 사용자와 시스템을 연결 시켜주는 기능을 한다. 사용자 입장에서는 시스템이 얼마나 사용하기 편리한가를 결정 해주는 것이, 가장 큰 요소임으로 그 중요성이 점점 더 크게 부각되고 있으며 graphics user interface가 사용되고 있다[12].

2.2. E.1의 특징

E.1은 구조설계용 지식베이스 개발용 도구이기 때문에 어느 정도 복잡한 수치계산도 지식베이스를 통하여 효율적으로 처리할 수 있다. 이것은 다른 지식베이스 개발용 도구에서는 찾을 수 없는 다음 두 가지 특징을 갖고 있기 때문이다.

1) 자동 반복 계산 기능(Automatic iteration process)

선급 규정이나 선박의 주요치수 결정을 위한 지식베이스를 만들 때 다음과 같은 rule형태가 흔히 발생한다.

```
rule1: if <condition1>
      then Y=f(X,...);
rule2: if <condition2>
      then X=g(Y,...);
```

이 예에서 만일 주어진 목표(goal)가 Y라고 하면 E.1의 추론 기구는 Y값을 찾기 위해 'rule1'을 검사한다. 여기서 'rule1'의 <condition1>이 참값이라고 가정 하면 E.1의 추론 기구는 Y값을 계산하기 위해 X값을 찾게된다. 여기서 다시 새로운 목표(goal)로 X가 주어지게 되고 'rule2'의 <condition2>가 참값을 갖는다고 할 때 X값을 계산하기 위해 또 다시 Y값을 요구하게 된다. 따라서 일련의 추론계산 과정은 무한 loop를 돌게 된다. 그러나 E.1의 추론 기구는 Y값을 결정하기 위해서 X나 Y 두 변수 중 하나를 선택해 초기값을 설정하여 반복 계산을 수행해야 함을 감지하고 자동 반복계산 기능(automatic iteration process)을 부르게 된다. 자동 반복 계산 기능은 사용자에게 이 사실을 알

리고 서로 연계된 변수중 하나를 선택해 초기값을 설정하도록 하고 원하는 정확도가 얻어질 때까지 반복 계산 과정을 수행한다. 이 과정이 끝나면 E.1은 Working memory를 반복 계산 과정이 시작되기 전의 상태로 만들어 놓고 계속해 추론하게 된다.

2) 재추론 기능

구조설계 과정 중 많은 부분이 반복적 의사 결정 과정(iterative decision making process)으로 되어 있기 때문에 이 과정을 효율적으로 뒷받침 해주기 위해서 만들어진 기능이다. 예를 들어 다음과 같은 rule들이 있다고 할 때(Fig. 2참조) rule 0까지 추론이 진행되었다고 가정하자. 이때 ruleA와 ruleB에 포함되어 있는 변수X의 값만이 새로운 값으로 대체 되었을 경우 X의 값에 따라서 ruleA나 ruleB 또는 그 상위 rule들은 어떤 새로운 결론에 도달하게 될 것이다. 이 새로운 결론을 알고자 할 때 E.1의 재추론 기능을 사용하면 된다. 이 기능을 수행하는 함수가 'eval'이다. 가령 rule0에 도달하게 될 새로운 결론을 얻고 싶으면 'eval' 함수를 사용하여 rule0를 재추론하면 된다. 물론 이때 rule 0의 하위 rule 1, rule 2도 X값에 따라서 새로운 결론을 얻게 된다. 'eval' 함수는 지식베이스 내에서 E.1의 내장 함수로써 사용할 수 있고 또 E.1의 시스템 수준(level)에서 시스템 명령어로 사용할

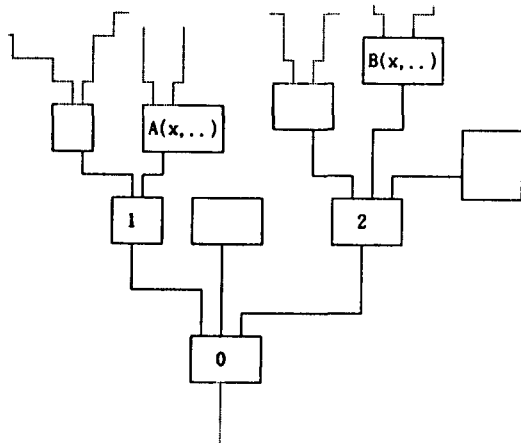


Fig. 2 Network between knowledge-base

수도 있다.

3. 지식표현(Knowledge representation)

E.1의 지식베이스 구문(syntax)은 rule, fact, table 등의 3 가지로 구분된다[13].

3.1 Rule 구문

Production rule 형식으로 Attribute-value쌍의 IF-THEN 형태이며 지식베이스 구문 중 가장 많이 쓰인다. 간략화 된 Bakus-Naur Form(BNF)를 사용하여 구문을 표시하면 다음과 같다[14] [15].

```
<rule>::<name>: if <premise>
      then <conclusion>;
<premise>::<expr> {and! or <expr>}
<conclusion>::<var> {=<expr>! <function>}
             {and<var> {=<expr>! <function>}}
```

이 구문을 사용하여 LLOYD규정 [16] 내용 중 chapter 1의 section 5에 있는 Table 1.5.2를 지식베이스화 한 예를 들면 아래와 같다.

```
%%% Chapter 1 Section 5
%%% Table 1.5.2
t_1_152_1:
if bottom_frame_type='longitudinal
then bottom_thick=
  max(0.001*s_t152_1*(0.043*L_t152_1
    + 10.0)*sqrt(FB/k),
    0.0063*s_t152_*sqrt(T*k/(2. - FB)));
%(mm)
```

3.2 Fact 구문

Fact 구문은 주어진 조건이 만족된 후 그에 따른 결론을 얻는 production rule형식과는 달리 있는 사실 그대로를 기술하는 것이다. Fact의 다른 형식으로 meta-fact가 있다. 현재 E.1은 3종류의 meta-fact 즉 message, list, watch를 갖고있다. E.1은 message를 사용하여 필요시 사용자에게 어떤 변수의 값을 묻게 될 때 message에 있는 문자열(string)을 출력하고 입력된 값의 type를 검사한다. List는 변수가 스칼라양이 아닌 벡터양임을 나타낸다.

<p>L, D, T, s, S, k as defined in 1.5.1</p> <p>$s_1 = s$ but is not to be taken less than $470 + \frac{L_2}{0.6}$</p> <p>$L_1 = L$, but need not be taken greater than 190 m</p> <p>$L_2 = L$, but need not be taken greater than 215 m</p>	<p>$F_g =$ as defined in Pt 3, Ch 4, 5.7</p> $f_1 = \frac{1}{1 + \left(\frac{s}{1000S}\right)^2}$ <p>$R_g =$ bilge radius, in mm</p>
--	--

즉 한 변수가 여러 값을 동시에 가질 수 있다는 뜻이다. Watch는 지식베이스를 추론시 지식베이스에 있는 어떤 변수들의 값의 변화를 알고 싶을 때 그 변수들의 값을 Watch window에 출력 하도록 E.1에게 알려 주는 역할을 한다.

이들을 BNF 형식으로 기술하면 아래와 같다.

```
<fact>::<factname> : <var>=<expr>
      <var>;
```

```
<list>::list(<var> {<var>});
```

```
<message>::message(<var>, <type>,
      (<string>{<string>}));
```

LLOYD 규정 Chapter 1 Section 5의 Table 1.5.2의 일부를 예로 들면 다음과 같다.

```
t_1_152_5 : s_t152_1
      =max(bottom_long_spacing,
      470.0+L_t152_2/0.6);
t_1_152_5 : s_t152_1=min(L, 190.0);
```

.....

```
message(bilge_radius, real, ("What is the value
      of radius of bilge ?"));
```

```
watch(3, s_t151, s_t152, f_t152_1);
```

E.1의 추론 기구가 bilge_radius의 값을 찾고자 할 때 먼저 지식베이스를 찾아보고 bilge_radius의 값을 결정할 수 있는 rule이나 fact가 없으면 사용자에게 그 값을 묻게 된다. 이때 bilge_radius에 관계된 meta-fact인 message가 존재하는 가를 검사한다. 위의 예에서는 bilge_radius의 message가 지식베이스에 있기 때문에 E.1의 추론 기구는 message의 세번째 parameter인 문자열(string)을 사용자에게 출력하고 입력을 기다린다. 사용자가 입력을 하면 그 값의 type을 검사하는데 위의 예에서는 message의 두번째 parameter가 "real"이기 때문에 사용자가 입력한 값이 실수(real

number)인지 아닌지를 검사한다.

만일 실수(real number)가 아니면 E.1의 추론 기구는 잘못을 알리고 다시 사용자에게 입력을 요구한다. 사용자가 올바른 입력을 할 때까지 이 과정을 반복한다. Watch는 E.1에게 세 변수 s_t151, s_t152, f_t152_1의 값이 변할 때마다 그 값을 Watch window에 출력하도록 하는 역할을 한다.

3.3 Table 구문

E.1만이 갖는 지식베이스 구문으로 table이 있다. 흔히 선급 규정은 필요한 변수들의 값을 표를 사용하여 나타내는 경우가 있다. 그런데 만일 이런 표를 production rule형식으로 표현하게 되면 그 표현이 복잡하여 이해하기 어려울 뿐만 아니라 rule의 양이 크게 증가한다. 이러한 문제점을 해결하기 위하여 table구문을 만들게 되었다. table 구문을 사용하면 이러한 표는 그 형식 그대로 지식베이스화 할 수 있다. 또한 table구문은 변수의 값들이 graph로 표시되어 있는 경우도 매우 효율적으로 graph를 지식베이스화 할 수 있다.

table구문을 BNF로 표시하면 아래와 같다.

```
<table>::name : <string>
      return : <return_value>
      size : <integer>, <integer>
      row_key : <row_keys>
      column_key : <column_keys>
      <table_body>;
```

이 table구문 사용 예를 들면

KR 규정[17] 제 3편 6장 제 1003조의 표 6.31을 production rule형식으로 표현하면

```
trule1 : if L<30.0
      then V_over_sqrt_L=1.77;
```

```
trule2 : if L >= 30.0 and L < 40.0
then V_over_sqrt_L
= 1.77 - (30. - L) / (30. - 40.) * (1.77 - 1.66) ;
trule3 : if L > 30.0 and L <= 6.0
then V_over_sqrt_L
= 1.66 - (40. - L) / (40. - 60.) * (1.66 - 1.31) ;
```

이와 같이 복잡한 production rule형식의 표현을 table 구문을 사용하여 표현하면 다음과 같이 간단히 표현된다.

```
table(name : "table 6.31 for V/sqrt(L)"
size : 1, 6
return : V_over_sqrt_L
row_key : null
column_key : interpolation
```

L"
30.0	1.77
40.0	1.66
60.0	1.31
80.0	1.40
100.0	1.33
120.0	1.30

선박의 주요치수를 추정하기 위해서 주기관의 마력을 추정할 필요가 있다. Admiralty계수(Cad = 배수량^{2/3} * 속도³ / 주기관 마력)를 사용하여 주기관 마력을 추정할 수 있는데 Cad의 값은 기준 선형(Cd=0.8, L/B=6.0)에 대한 값에 설계하는 배의 L/B와 Cb가 차이가 있을 경우 그 수정치를

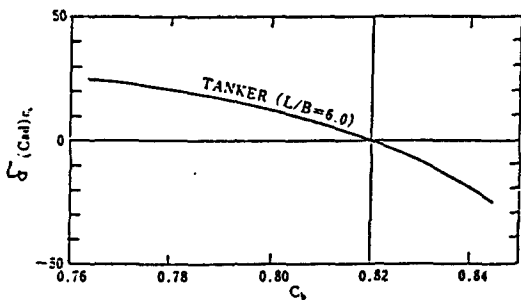


Fig. 3 Correction of cad for cb

더해 주어야 한다. Cb의 차이에 따른 수정치는 다음과 같이 Fig. 3으로 표시된다[18].

Fig. 3을 table구문을 사용하여 지식베이스화 하면 다음과 같이 간결히 표현된다.

```
table(name : "Fig.3"
size : 1, 7
return : correction_for_Cb
row_key : null
column_key : interpolation
```

"Cb"
0.76	25.0
0.78	20.0
0.8	12.0
0.82	0.0
0.83	-10.0
120.0	-20.0
0.8405	-26.0

```
);
```

4. 살물선의 초기설계 및 중앙단면 설계용 지식베이스 전문가 시스템

4.1 지식베이스 전문가 시스템의 구성

시스템의 전체적인 구성을 간략히 도식화한 것이 Fig. 4이다.

선박의 주요치수를 추정한 후 그 결과를 사용하여 중앙단면을 대략적으로 설계하게 된다. 이때 중앙단면의 기본형상(이중저 높이, hopper tank depth, topside tank depth, bracket spacing, keel plate length,...)을 결정해야 하는데 이 단계에 해당하는 지식베이스는 아직 개발되어 있지 않다. 그래서 설계자는 미리 유사선의 실적 자료나 과거의 설계 경험을 통하여 필요한 자료를 준비하여야 한다. 중앙단면 설계에 필요한 모든 데이터가 시스템에 입력되면서 설계가 진행되는데 이때 설계자는 E.1의 watch기능, 재추론 기능, graphics 기능을 사용하여 중앙단면 형상을 결정하는데 많은 도움을 받을 수 있다. 이 단계에서 얻은 결과를 사용하여 최적화 program의 입력 file을 생성하고

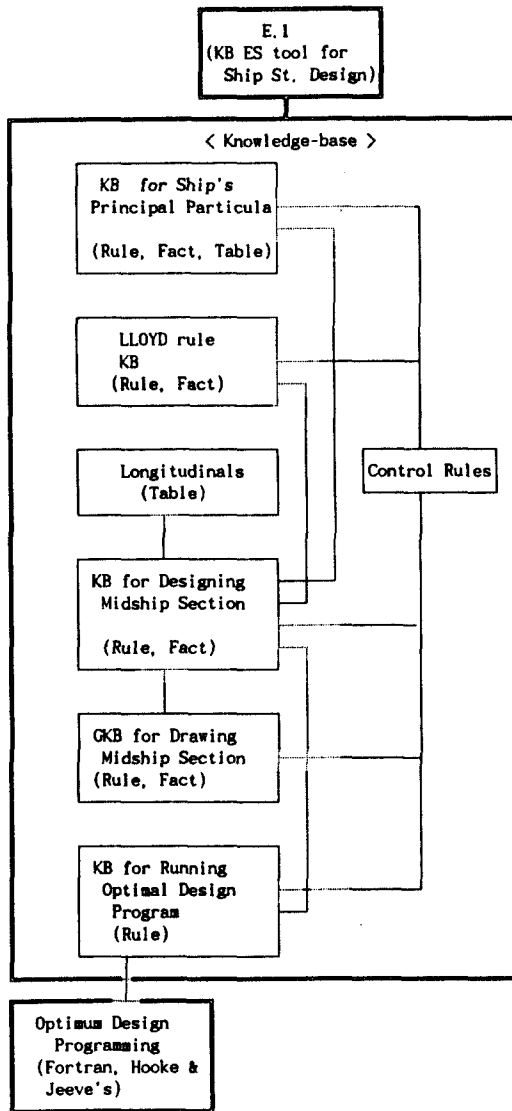


Fig. 4 Knowledge-based expert system for ship structural design

중양단면 최적화 프로그램을 수행시킴으로써 일련의 설계 과정을 끝 마치게 된다.

1) 선박의 주요치수 결정을 위한 지식베이스(KB)

Fig. 4의 "선박의 주요치수 결정을 위한 KB"는 선주의 요구조건으로 선박의 속도, 흘수, 재화중량 등이 주어지면 이 조건을 적절히 만족시켜줄 선박의 수선간 길이, 형폭, 형심, 방형계수 등을 추

정하기 위해 사용되는 지식베이스이다. 이 과정을 아래와 같이 design spiral로 표시할 수 있다 [16].

- a). Displacement 추정
 - b). 수선간 길이, 방형계수, 형폭, 형심 추정
 - c). 주기관 회전수 및 마력 추정
 - d). 경하 중량 추정(선각 중량, 의장 중량, 기관부중량)
 - e). 재화중량 검사
 - f). 주요치수를 수정하고 재화중량 검사 과정이 만족스러울 때까지 위의 과정을 계속 반복
- 위의 design spiral 과정은 Control rule로써 지식 베이스화 하였고 design spiral의 f) 과정은 E.1의 재추론 기능을 사용하여 구현되었다. 주요치수 결정을 위한 경험식들의 설계 변수들은 서로 network를 구성하고 있으며 입력 데이터와 설계 변수와의 관계는 rule, fact, table을 사용하여 지식 베이스화 하였다.

2) 중양단면 설계용 지식베이스(KB)

Fig. 4에서 "LLOYD rule KB"는 선박의 중양단면의 부재에 적용되는 부분을 뽑아서 만든 것으로 중양단면 설계외에 다른 용도로 사용될 수 있는 일반성을 갖고 있다. 종방향 보강재(longitudinal)는 table 구문을 사용하여 필요한 치수들을 지식 베이스화 하였다.

Fig. 4에서 "LLOYD rule 적용 및 중양단면 설계용 KB"는 LLOYD rule을 적용하기 위한 가정과 계수들을 구하는 과정, 중양단면의 기본형상을 결정하고 단면계수를 구하는 과정을 지식베이스화 한 것이다. 특히 이 부분은 일반적으로 체계화 되어 있지 않은 부분이기 때문에 지식베이스가 어떤 출처(source)를 중심으로 작성되었는가에 따라서 설계결과가 크게 차이를 보일 수 있다[16] [19] [20].

중양단면을 구성하는 부재는 모두 21개라고 가정하고 이 모든 부재들의 property(두께, 길이, 면적, 부재단 좌표, 면적 모멘트, 관성 모멘트)들이 요구되는 모든 제한 조건을 만족시키도록 그 값을 결정하면 된다. 이때 설계될 수 있는 중양단면의

경우는 무수히 많은데 가장 바람직한 해는 중앙 단면의 면적이 최소가 되도록 하는 것이다. 이것을 위해 최적화 프로그램이 사용된다. 중앙단면 설계의 일반적인 과정은 각 부재의 property를 계산하고 요구되는 deck나 bottom에 대한 단면계수가 만족되는지를 검사하고 그렇지 않은 경우 일부 설계 부재의 property(값 (deck의 두께,...)를 적절히 수정한 후 요구되는 제한조건이 만족될 때까지 이 과정을 반복한다. 이것이 Control rule로써 E.1의 재추론 기능을 이용하여 지식베이스화 되어 있다.

최종적으로 모든 부재의 property가 결정되면 “중앙단면 형상을 그리기 위한 KB”를 통하여 중앙단면을 구성하고 있는 판(plate)과 보강재(longitudinal)의 형상을 그리게 된다.

3) 중앙단면 최적화

중앙단면 최적화 프로그램은 Fortran으로 작성되어 있으며 최적화 기법은 HOOKE & JEEVE'S를 사용하고 있다. 지식베이스의 추론으로 얻은 결과를 입력 file로 만들고 최적화 프로그램을 수행 시키게 된다. 이 과정을 지식 베이스화 한 것이 Fig. 4에서 “최적화 program의 수행 및 입력 data를 만들기 위한 KB”이다.

4.2 수행 결과 및 고찰

선주가 제시한 조건이 Table 1과 같을 때 전체 시스템의 수행 과정 및 결과를 살펴 보았다.

Table 1. Owner's requirement

Type	: Bulk Carrier
Deadweight	: 37000ton
Speed	: 13.7 knot
Draft	: 11.0m

1) 선박의 주요치수 결정

Fig. 4에 있는 “선박의 주요치수 결정을 위한KB”를 추론하여 요구되는 선박의 주요치수를 결정하는 과정이다.

Fig. 5는 전체 design spiral 과정 중 f) 항에 해당되는 과정을 보여주고 있는 데 주어진 재화중

량과 추정된 재화중량간에 큰 차이를 보이면 지식베이스 내에서 E.1의 재추론 기능을 호출하여 만족된 결과가 나올때까지 주요치수를 수정하고 design spiral과정을 반복 수행하게 된다.

주요치수의 수정은 적절하다고 제시된 Lpp, breadth, Cb 중 하나 혹은 그 이상의 것을 선택해서 선택된 변수의 값을 E.1의 시스템 명령어인 “set”을 사용하여 수정한다. Fig. 5에서는 Lpp의 값만 수정하였다.

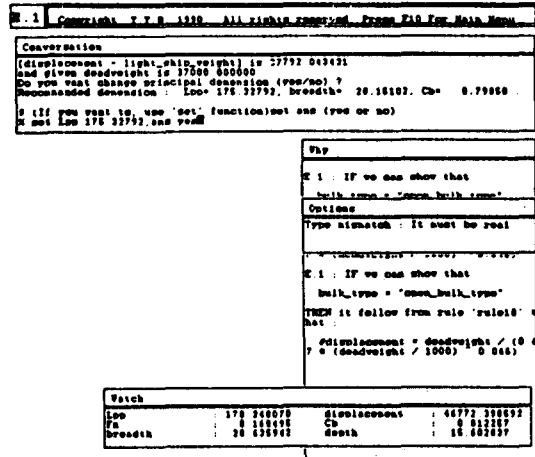


Fig. 5 Example of re-inference process in E.S. for preliminary design

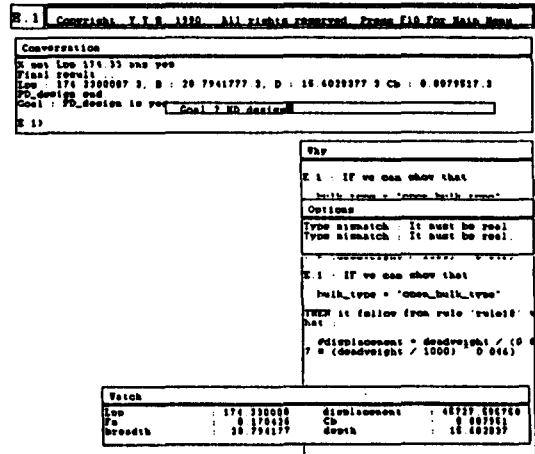


Fig. 6 Final result of preliminary design

Fig. 6은 최종적으로 얻은 주요 치수를 Conversation window에 출력하고 있다. 산출된 주요치수는 Table 2와 같이 결정되었고 이 값을 이용하여 중앙단면의 설계과정으로 넘어간다.

Table 2. Principal particulars(m)

Lpp	174.33
Breadth	28.79
Depth	15.60
Cb	0.8080

2) 중앙단면 설계

중앙단면의 기본형상을 결정하기 위한 지식 베이스는 개발되어 있지 않기 때문에 유사선의 자료나 설계자의 지식을 통하여 Table 3과 같은 데이터를 준비해야 한다.

Fig. 7은 E.1의 자동 반복 계산 기능 예를 보여 주고 있는 데 FB (bottom에 대하여 요구 되는 단면계수/bottom에 대한 실제 단면계수)의 값을 결정하기 위해서는 반복 계산이 필요함을 알리고 연계된 변수 중의 하나를 선택해서 초기값을 입력하도록 설계자에게 요구하고 있다.

Fig. 8은 deck에 대한 단면계수의 값이 요구치 보다 적을 경우 deck의 두께를 변화 시킨후에 재추론 기능에 의해서 다시 추론을 수행하는 과정을 보여주고 있다.

Table 3. Basic information of infra structures in midship section

top web space : 3.75(m)	topside tank depth : 4.54(m)
floor space : 2.25(m)	camber : 0.8(m)
frame space : 750.0(mm)	kunckle point : 7.2(m)
stowage rate : 0.694	duct girder distance from center girder : 1.6(m)
number of side girder : 3	
double bottom height : 1.7(m)	topside tank bracket : 1.2(m)
bilge radius : 2.0(m)	hopper tank bracket : 1.6(m)
hopper tank height at side : 3.240(m)	hopper tank breadth at inner bottom : 3.8(m)
hatch coaming : 0.318(m)	longi. spacing : Fig.9 refer

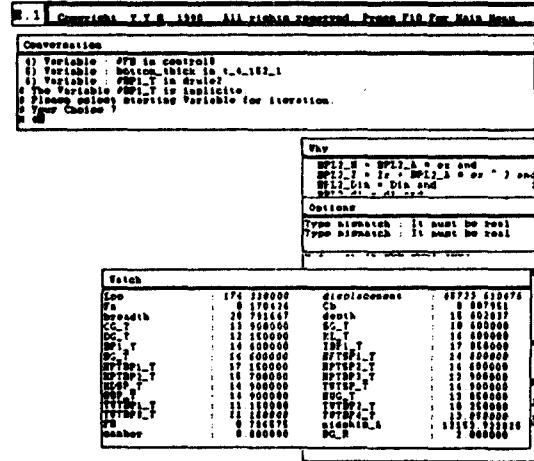


Fig. 7 Example of automatic iteration process in E.S. for midship section design

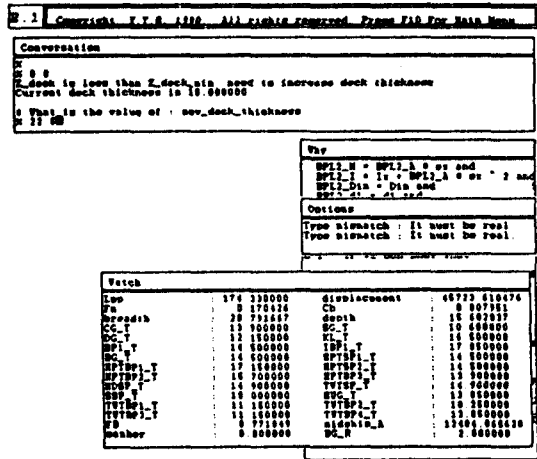


Fig. 8 Example of re-iteration process in E.S. for midship section design

Fig. 9는 설계된 중앙단면의 형상과 각 부재의 치수를 보여 주고 있다. "set" 명령어를 사용하여 bilge의 반지름을 2.0m에서 1.7m로, double bottom의 높이를 1.7m에서 2.0m로 변화 시킨 후 재추론 기능을 사용하여 다시 추론하려 하고 있다.

Fig. 10은 변화된 bilge의 반지름과 double bottom의 높이에 따른 새로운 결과를 보여 주고 있다.

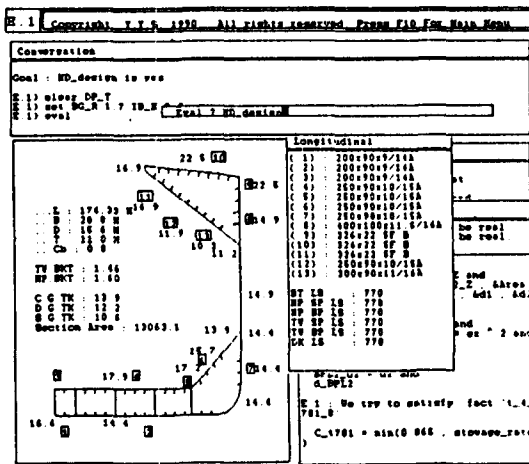


Fig. 9 Result of midship section design
(Case : bilge radius = 2.0
double bottom height = 1.7)

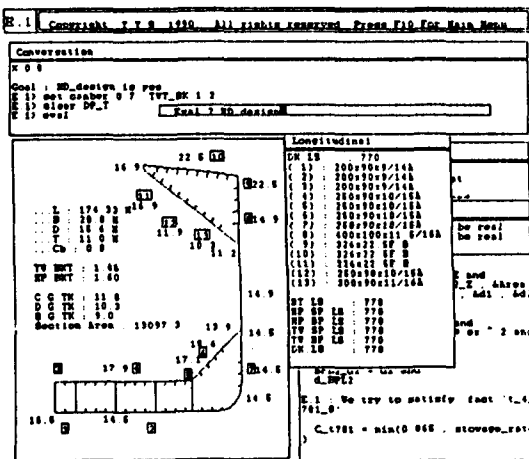


Fig. 10 Result of midship section design
(Case : Bilge radius = 1.7
double bottom height = 2.0)

또 다시 중앙단면의 형상을 변화 시킨후 재추론을 하려는 것을 보여 주고 있다. 이 예들에서 알 수 있듯이 Watch window, 재추론 기능을 이용하면 중앙단면의 각 부재의 설계뿐만 아니라 기본형상을 결정하는데 설계자에게 큰 도움을 줄 수 있다.

3) 중앙단면 최적화

설계된 중앙단면의 기본 데이터(Fig. 10)를 사용하여 최적화 프로그램에 필요한 입력 file을 만든 후 최적화 프로그램을 수행시키게 된다. 최적화 프로그램의 수행 결과에서 얻은 중앙단면의 면적은 Fig. 10의 중앙단면의 면적과 비교하여 약 1.7% 정도 감소하였다. 최적화 프로그램의 수행 결과는 Table 4와 같다.

Table 4. Optimization results of midship section

	OPTIMUM VALUE(mm)
DECK LONGL. SPACE	935.0
DECK LONGL. THICKNESS	21.5
DECK LONGL. HEIGHT	385.0
BOTTOM LONGL. SPACE	770.0
HOPPER LONGL. SPACE	765.0
SIDE LONGL. SPACE IN HOPPER TANK	770.0
TOP SIDE PLATE LONGL. SPACE	775.0
SIDE LONGL. SPACE IN TOP SIDE TANK	810.0
*** TWBKT SIZE	= 1479.3mm
*** HPBKT SIZE	= 2180.0mm
CROSS SECTIONAL AREA	: 12877.5cm ²

5. 보강판 설계를 위한 지식베이스 시스템

구조물의 설계에 있어서 설계조건을 만족하면서 최소의 비용이 들도록 설계변수를 결정하는 것이 과제이다. 이때 설계 기술자의 경험이 크게 좌우되나 보통 최적 설계법이 많이 사용된다. 그러나 최적 설계법은 일반적으로 복잡한 해석 과정이 많이 수반된다. 따라서 엄밀한 최적해를 요구하는 경우가 아니라면 계산시간이 적게 들고 간편히 설계할 수 있는 실용적인 방법을 설계자는 요구하게 된다. 그래서 본 연구에서는 최적화 방법에 의한 결과를 회귀분석하여 얻어진 지식베이스를 이용함으로써 설계자가 간단히 실용적으로 사용할 수 있는 간이 설계법을 모색하여 보았다[21].

5.1 보강판 설계를 위한 지식베이스

설계하중은 면내하중 p와 면외하중 q가 있다. 면내하중이 작용하는 변을 A, 면외하중이 작용하지

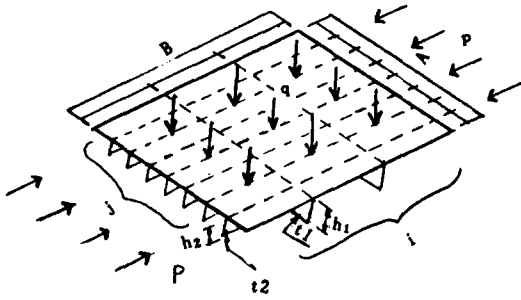


Fig. 11 Geometry of stiffened plate

얇은 변을 B라고 칭하고, 판의 두께를 t로 한다. (Fig. 11 참조) 변A에 평행하게 폭이 h1, 두께가 t1인 보강재가 i개가 있고, 변 B에 평행하게 폭이 h2, 두께가 t2인 보강재가 j개가 있다. 항복응력은 25kgf/(mm²), Young률은 21000kgf/(mm²), 포와 송비는 0.3으로 가정하였다. 이때 결정해야 될 설계변수는 t, i, h1, t1, j, h2, t2 인 7개가 된다. 변의 크기 A, B는 2000mm에서 10000mm까지 2000mm 간격으로 조합하여 25가지인 경우로 나누고 설계하중 p는 5 - 25kgf/(mm²), q는 0.005 - 0.025kgf/(mm²)의 범위에 한하여 보강판의 중량을 최소로 하는 최적화 program을 수행시켜 그 결과를 얻었다. 이 결과를 사용하여 각각의 설계변수들을 다음과 같은 p, q의 함수로 나타낼 수 있다고 가정하여 회귀분석을 수행하여 설계변수에 대한 지식베이스를 구성하였다.

$$\hat{n} = A\phi + A1 * p + A2 * q + A3 * p * q + A4 * p^2 + A5 * q^5 + A6 * P^2q * + A7 * p * q^2 + A8 * (p * q)^2$$

따라서 25*7개의 if-then rule 형식의 지식베이스가 얻어지게 되는데 이때 지식 베이스의 크기를 줄이기 위하여 table구문을 사용하였다. 이것을 그림으로 나타낸 것이 Fig. 12이다.

이 지식베이스를 사용하여 설계변수를 결정했을 때 최적해에 근사하는 정도는 앞에서 가정된 함수에 따라 좌우된다. 최적해의 값이 일정한 경향을 보이기 보다는 분산된 형태를 취하기 때문에 정

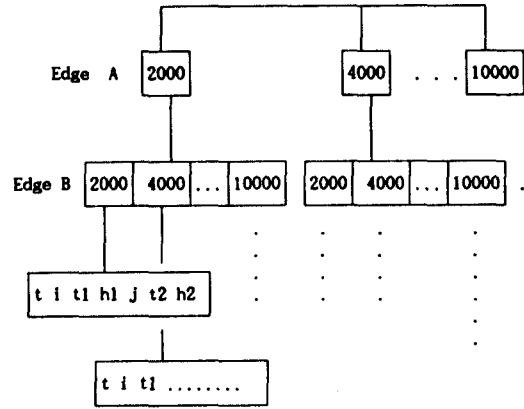


Fig. 12 Knowledge-base for design variables of stiffened plate

확한 값을 주는 함수의 선정은 어려운 문제이다. 예를 들어, 판의 크기가 A=3000mm인 경우, 위의 지식베이스에 직접 적용할 수 없을 때는 보간법을 사용한다. 이 지식베이스에서 결정된 설계변수들이 제한조건을 만족하는지를 검사하기 위하여 판, 보강재, 보강판의 좌굴하중과 붕괴하중을 결정하기 위한 식들[20]이 지식베이스화되어 있다. 대부분의 경우 제한조건을 만족하지만 그렇지 않은 경우나 제한조건을 만족하더라도 보강판의 중량을 줄이기 위해서는 설계변수들의 값을 수정하고 E.1의 재추론 기능을 사용하여 다시 설계하게 된다. 이 시스템에서는 적절한 설계변수를 선택하여 어느정도 변화시켜야 하는지를 설계자에게 알려주는 기능은 아직 없다.

5.2 수행 결과

Table 5에서 보는 바와 같이, 임의의 4 경우에 대해 지식베이스를 이용하는 간이설계법의 결과와 최적해의 결과를 비교하면 약 20%범위내의 보강판 설계가 얻어지고 있음이 확인되었는데, 이는 최적화 과정을 이용하지 않고 설계를 하는 경우 만족할만한 결과로 생각된다. 물론 지식베이스를 구성하기 위해 사전에 25*7개 경우의 최적해를 구하는 과정이 선행되어야 하지만, 지식베이스가 수치계산을 통해서든지, 또는 실적선의 자료를

Table 5. Results obtained by regression formula for stiffened plate

CASE	A(mm)	B(mm)	p(kgf/mm ²)	q(kgf/mm ²)	Vol(mm ³)	Opt.	Error(%)
1	2000	6000	5	0.01	5.480	5.620	2.5
2	6000	4000	15	0.015	11.325	11.656	2.8
3	8000	10000	15	0.02	18.131	15.075	20.
4	10000	10000	20	0.02	26.515	24.370	8.8

$$\text{Where Vol} = (A \cdot B \cdot t + i \cdot t_1 \cdot h_1 + j \cdot t_2 \cdot h_2) / (A \cdot B)$$

$$\text{Error} = | \text{Opt} - \text{Vol} | / \text{Opt} \cdot 100(\%)$$

이용하는 경험적인 방법을 통해서든지 간에 형성되기만 하면, 이 지식 베이스를 이용하여 설계를 간단히 할 수 있음을 검토하였다. 또한 이 방법은 엄밀한 최적해가 필요한 경우는 지식베이스에 구한 설계변수를 기준으로 하여 탐색 범위를 결정할 수 있기 때문에 최적화 program 수행시 필요없는 탐색 범위를 배제시킬 수 있으므로 program 수행 효율을 높일 수 있다.

6. 결 론

1) 전문가 시스템 개발용 도구인 E.1을 C 언어로 사용하여 개발하였고, 설계과정상 필요한 자동반복 계산과 재추론 기능을 추가하여 추론과정에 있어서 E.1의 효율성을 높였다.

2) 지식베이스의 구성에 있어서 통상 사용되는 IF-THEN 형식의 Production rule 방법 외에도 FACT구문과 TABLE구문을 사용하여 지식표현 방법을 다양화 하였으며, 특히 TABLE 구문은 불필요한 지식베이스 표현을 단순화 할 수 있어 지식베이스 양을 줄일 수 있었다.

3) 보강판 설계에 있어서 기존의 최적화 프로그램을 이용하여 얻은 최적치 결과를 토대로 회귀분석하여 유도된 간단한 설계식을 지식베이스화 함으로서 복잡한 구조설계에 있어서 전문가 시스템을 이용한 간이설계법의 가능성을 확인하였다.

4) E.1의 지식표현은 크기가 작은 지식베이스의 개발에는 큰 문제점이 없지만 지식베이스의 양이 클 경우에는 지식의 유지 관리 면에 문제가 발생

하게 될 것이다. 지식베이스의 양이 클 경우 변수들을 IF-THEN 형식만으로 나타내기에는 비효율적이어서, 앞으로 지식베이스의 계속적인 개발과 축적한다는 관점에서 볼 때 frame방식이나 객체지향(object-oriented)된 지식표현 방식으로 전환할 필요가 있다.

후 기

본 연구는 해사 기술 연구소의 89년도 CSDP관련 연구 지원으로 수행 되었으며, 이 기회에 관계자 여러분께 깊은 사의를 표한다.

참 고 문 헌

- [1] Rasdorf, W.J., "Generic Design Standards processing in an Expert system Environment", *Journal of computing in Civil engineering*, Vol. 2, No.1, Jan, 1988
- [2] Dym, C.L., "Application of Knowledge-based system to Engineering Analysis and Design", *ASME*, 1985
- [3] Waterman, H-R, "Building expert system", Addison-Wesley, 1983.
- [4] Tomiyama, T., & Yoshikawa, H., "Requirement and Principles for Intelligent CAD System", IFIP WG 5.2 Working Conference on Knowledge Engineering in Computer-Aided Design, 1985.
- [5] Ohsuga, S., "Conceptual Design of CAD Sys-

- tems Involving Knowledge Bases", IFIP WG 5.2 Working Conference on Knowledge Engineering in Computer-Aided Design, 1985.
- [6] 伊藤健, "선박 초기 설계용 대형 전산 시스템-MMARINE" 日本造船學會誌, 1987, 5.
- [7] Ohstsubo, H., "Structural Design of Midship Section by Expert system", 日本造船學會誌, 1988, 11.
- [8] RaviSethi, A. V. & Ullman, J. D., "Compilers Principles, Techniques and Tools", Addison Wesley, 1985.
- [9] Sahni, E. H. S., "Fundamentals of Data Structures in Pascal", Computer Science Press.
- [10] Gough, K. J., "Syntax Analsis and Software-Tools", Gough
- [11] Fischer C. N. & LeBlance, R. J. Jr., "Crafting a Compiler", The Benjamin/Cummings Publishing Company, Inc.
- [12] 양영순 "CSDP 중앙단면 최적화에 있어서 AI기법 응용", 1989, 9
- [13] 연윤석 "조선공학에 이용 가능한 Knowledge-base shell, E.1의 개발 및 고찰", not published, 1988.
- [14] 이경호 "중앙 단면 최적설계를 위한 전문가 시스템 개발에 대한 연구", 서울 대학교, 석사 졸업 논문, 1989.
- [15] Adeli, H., "Expert System in Construction and Structural Engineering", Chapman and Hall Ltd, 1988.
- [16] Lloyd's Register of Shipping, "Rules and Regulations for the Classification of Ship", 1985.
- [17] 한국 선급. "강선 규칙", 1985
- [18] 이규열, "서울대학교 선박 계열 강의 note", 1989
- [19] 함주혁, 강점문, "Computer Aided Midship Section Design of Bulk Carrier Based on L.R. Rule and H.H.I.Practice", 技術現代, 1989
- [20] 윤장호, 김현권, "船舶最適構造設計(II) (TANKER의 最少重量設計 PROGRAM 開發)", C-032完了 報告書, 船舶海洋研究所, 1987, 11.
- [21] Aoki, G., "Optimum Design of Stiffened Plates using the Object-Oriented Programming Language SMALLTALK", 船舶技術研究所 報告 제 25 권 제 6 호, 1988.