

## 바둑에서 형세판단에 근거한 효율적인 정석 선택\*

김 경 아\*, 박 현 수\*, 이 중 철\*\*

### Effective Selection of Jeongseok Based on States Evaluation in Baduk

Kyung-A Kim, Hyun-Soo Park, and Chong-Cheol Lee

**Abstract** In order that we can choose effectively one jeongseok from many alternatives in the earlier stages of Baduk games, we adopt a model of state evaluation that is similar to the traditional one, where we guess the boundaries that are not yet complete, as if one who has the higher priority could do as it wish to. In that model, we split the cases into three parts, namely, one in which one side has the higher priority, one in which both sides have the same higher priority, and the one in which both sides have the same lower priority. We represented the jeongseoks, namely whajeom, gomok, somok, and samsam jeongseoks, by way of trees, and we proposed an algorithm that chooses the most favorable jeongseok, using the results of state evaluation.

#### I. 서 론

인공지능 분야는 광범위하고 실제적인 문제들을 해결하는데 있어 인간이 사고하는 복잡하고 지적인 과정에 어떻게 하면 더 가깝게 근접할 수 있을지를 목적으로 한다. 주로 패턴인식, 전문가 시스템, 컴퓨터 비전과 같은 여러 응용 분야에서 눈부신 발전이 있어왔지만 특히 게임이론분야는 사람과 비교하면 아직도 많은 차이를 나타내고 있는 것이 현실이다. 두 사람이 완전한 정보를 가지고 하는 게임분야중에서 많은 연구가 진행되고 있으며 사람에 근접한 지능을 가지는 체스에 비해 그 규칙은 유사하지만 고도의 지적인 인간 사고력을 요구하는 바둑게임의 발전은 미흡한 편이다.

바둑판의 크기와 게임의 특징으로 인해 다른 게임에서 사용해 온 게임 트리의 탐색 방법의 적용은 비효율적이 되었다. 그래서 지금까지 연구는 단순한 탐색중심의 방법보다는 knowledge 데이터베이스를 사용하거나 부분적으로 전략적인 접근방법을 사활, 형세판단, 끝내기에 적용해 왔다<sup>1~7)</sup>. 체커의 게임트리가  $10^{40}$ 개, 체스는  $10^{120}$ 개의 노드를 가지는데 비해 바둑은  $10^{700}$ 가지 경우가 발생할 수 있다고 한다.

바둑 게임은 19\*19크기의 빈 바둑판에 두 사람이 교대로 흑과 백돌을 놓게 되는데 상대방 돌로 포위된 돌은 딱딱하게 되며 두 사람이 모두 차례를 넘기면 게임은 끝이 난다.

이렇게 진행되는 바둑게임은 초반, 중반, 중반 3단계로 나누어 지는데, 결국은 중반에 집을 많이 차지하는 편이 이기는 지적게임이다.

게임의 시작단계인 초반에 형세를 유리하게 이끌어 가는 것이 무엇보다도 중요하다 하겠다. 초반은 포석 및 정석단계로 귀에서부터 착점을 하며 귀와 연결된 변으로 벌려나가면서 유리한

1994년 12월 31일 접수

\* 경북대학교 대학원 컴퓨터공학과 석사과정

\*\* 경북대학교 공과대학 컴퓨터공학과 부교수

(※ 본논문은 1994년도 한국학술진흥재단의 공모 과제 연구비의 부분적인 지원에 의하여 연구되었음.)

곳에 착점을 한다.

정석이란 귀에서 발생되는 모양의 수순을 기록해 놓은 것으로서 그 상태에서는 서로간에 최선의 응수임을 의미한다. 귀에서 정석으로는 화점, 소목, 고목, 외목, 삼삼정석과 변, 중앙에 대해서 기록되어 있으며 현재 알려진 것만 해도 10여만 가지가 있다. 이렇게 구성되어 있는 정석은 같은 것이라도 현재 국면에 따라 살아있는 정석이 될 수도 있고 또는 죽은 정석이 될 수도 있다.

게임을 진행하다 보면 임의의 상태에서 선택할 수 있는 정석은 여러 가지가 있을 수 있는데 어떤 정석을 선택할지 정확한 기준은 없다. 그러나, 전문기사의 경우를 보면 이제까지 쌓아온 경험을 바탕으로 한 나름대로 전략을 가지고 있다. 즉, 정석을 둔 결과를 머릿속에 그려놓고 전국적인 균형을 판단해서 가장 좋은 정석을 선택하는 것이다.

그리고 중반은 주로 전투 및 공격, 방어, 영토 확장을 하는 단계로 이러한 많은 전략중 가장 효율적인 하나를 선택하기 위해 형세판단이 사용되어진다.

마지막으로 끝내기 단계는 선수 끝내기, 역 끝내기, 후수 끝내기 등 유리한 전략을 형세판단의 근거에 의해 행할 수 있다.

형세판단이란 앞으로 일어날 변화를 미리 예상하는 수읽기를 포함하는 바둑 고도의 기술로서 영토의 윤곽이 어느 정도 결정된 모양의 집수를 세어서 어느 쪽이 유리한가를 판단하는 것이다. 전문 기사들은 거의 정확한 형세판단 능력을 갖추고 있다. 포석이나 중반 전투 단계에서 형세판단의 결과로 다음 수나 전략을 선택하게 된다.

본 논문에서는 정석을 선택해야 할 임의의 국면에서 형세판단 방법을 사용하여 현재 국면에서 가장 유리한 정석 선택방법에 대해 연구하였다.

정석의 선택에 따라 실리와 세력으로 나누어 가지거나, 실리를 서로 나누어 가지는 것으로

구분된다. 이러한 정석을 양선수, 양후수, 편선수를 이용하여 영역을 확정해서 계산한 다음 세력을 계산하는 형세판단 방법을 제시하였다.

II장에서는 형세판단에 필요한 기본적인 용어 정의와 논문에서 정의한 형세판단 방법과 전문기사의 형세판단 결과의 예를 통하여 비교하였으며, III장에서는 컴퓨터 바둑에서 정석 데이터 구성 방법과 표현방법, 정석 진행 방법, 가장 효율적인 정석을 선택할 수 있는 알고리즘을 제시하였고 IV장에서는 실험을 통하여 형세판단 방법과 상용 프로그램에서 영역 표현 방법을 이용하여 정석을 선택하는 알고리즘의 타당성을 보였고 V장에서는 결론을 맺었다.

## II. 용어정의 및 형세판단 방법

### 1. 개요와 용어 정의

초반 바둑의 한 예로 그림 1의 좌하변에 놓인 정석을 살펴보면 흑이 실리, 백이 세력을 가지지만 우하변에 놓인 흑이 백의 세력을 감소시키고 있으며 세력을 가진 돌들로부터 좌변의 백한 점이 고립되어 있으므로 결국은 백에게 불리한 정석이 된다. 이처럼 현재 국면을 형세판단 하지 않고 정석을 둔 결과 오히려 자신에게 불리한 형세가 되었다. 그러므로, 정석을 전체 형세에 맞게 선택함으로써 국면을 유리하게 이끌어 낼 수 있는 것이다.

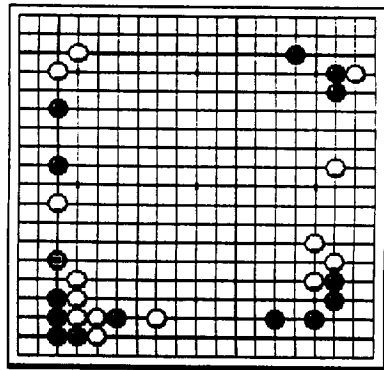


그림 1. 초반 바둑  
Fig. 1. Opening state of baduk.

형세판단에 있어서 가장 기본적인 단계는 목산인데 이것은 정해진 모양이나 정해진 형태가 몇집이다라고 판단하는 것이다.

확정지의 목산은 포석이 일단락 되었을 때, 한 부분의 전투가 끝나고 다음 전투에 들어갈 때, 끝내기예 들어갈 때 등 실전에서 되풀이 되는 것으로서 국면마다 다음 작전의 지침으로 사용될 수 있다.

본 논문에서는 먼저 확정지를 세고 다음으로 확정되지 않은 모양, 즉, 세력을 집으로 환산하는 두 단계를 형세판단의 기준으로 삼았다.

형세판단을 예로 들면, 양상국 6단이 실전대국의 초반전을 형세판단한 것을 그림2에 나타내었다. 흑집은 상변 38집, 좌변 7집, 우하귀 15집으로 합계가 60집이고 백집은 하변 26집, 우변 8집, 좌상귀 13집으로 합계가 47집이 되어 형세가 흑에게 유리하게 진행되고 있음을 알 수 있다. 그러므로 흑은 무리하지 않고 전체국면을 유지하는 전략을 선택 할 수 있는 것이다.

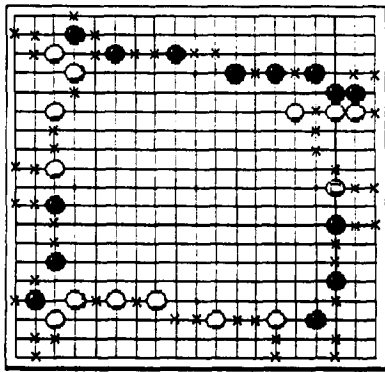


그림 2. 그림 1의 형세판단  
Fig. 2. State evaluation of fig. 1.

정석 선택을 위한 형세판단은 정석 특성상 초반에 발생되어진다. 초반에는 집의 형태가 분명하지 않으므로 먼저 끝내기할 때 처리되는 경계선을 예상해야 한다. 중반전의 끝내기는 이제까지 확보한 영토의 경계선을 마무리하는 단계로서 우선 어느 곳부터 할 것인가<sup>6)</sup>를 결정한 다음

선수 권리가 어느 쪽에 있느냐에 따라 경계선을 결정한다. 선수란 이쪽이 두면 상대방이 반드시 응수하는 수이다. 이러한 선수 개념에 따라 양선수, 양후수, 편선수 형태의 경계선을 정의하였고, 이 정의에 따라 돌과 변사이 링크를 확정한다. 초반의 특성으로 인해서 돌과 돌사이에 경계선을 결정하는 돌과 돌사이 링크의 위치는 사람이 형세판단하는 방법과 마찬가지로 그 위치가 주변 상황에 따라 유동적이어도록 하였다.

형세를 판단할 때 중요한 변수로서 세력을 정의할 수 있는데 이것은 어느 위치에 영향을 미치는 돌이 많으면 많을수록 그 위치는 그 돌의 영역이 될 확률이 높으므로 돌마다 방향에 따라서 빈자리에 미치는 세력을 계산하였다. 이러한 형세판단의 기본원리와 적용에는 참고문헌에 잘 설명되어 있다<sup>9 11)</sup>.

본 논문에서 사용되는 기본적인 용어는 이동기, 박현수<sup>5, 7)</sup>에 따르면 새로운 용어는 다음에 정의하였다. 바둑판 특성으로 인해 다음 정의는 함수  $f^a$ ,  $f_s$ 을 사용해서<sup>8)</sup> 90도, 180도, 270도 회전변환과 좌우대칭 변환에 의해 8가지가 같은 모양이 됨을 알 수 있다.

정의 2.1 돌과 돌사이 한 칸, 두 칸, 날일자, 눈목자 별림이면 링크(link)가 존재한다.

정의 2.2 돌과 변의 정점이 인접, 한 칸 별림이면 돌과 변사이 링크가 존재한다.

정의 2.3 돌과 돌 사이 링크가 존재하며 그 링크상에 적돌이 없는 같은색 돌들의 집합을 그룹(group)이라 하며 한 개 돌도 하나의 그룹이다.

정의 2.4 어느 한 쪽만이 선수의 권리를 가지는 경우를 편선수라 한다.

정의 2.5 양쪽다 선수의 권리를 가지는 경우를 양선수라 한다.

정의 2.6 양쪽다 후수인 경우를 양후수라 한다.

정의 2.7 임의의 그룹을 구성하는 돌을 정점(vertex), 돌과 돌사이 링크를 선분(edge)으로 그래프를 구성할 때, 다음 i), ii)가 이루는 영역을 집이라 정의하고 영역의 빈정점 수를 집 수라 한다. 단, 돌과 변사이 링크가 2개이상 존재할 때는 집 수가 더 많은 링크를 선택한다.

- i) 그룹 자체가 사이클(cycle)인 경우
- ii) 그룹과 양후수, 양선수, 편선수 형태의 돌과 변사이 링크와 그 링크에 속하는 변의 한 정점에서 같은 변이나 다른 변의 한 정점으로 연결 그래프로서 확정된 영역이 사이클을 이룰 때 빈정점 수가 더 적은 사이클인 경우

정의 2.8 그룹이 가지는 영향력을 세력이라 한다.

그림 3은 흑이 실리, 백이 세력을 가지는 대표적인 정석으로 본 논문의 정의에 따르면 흑은 9집, 백의 세력은 18집임을 표현하고 있다.

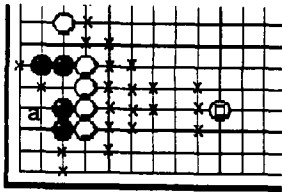


그림 3. 정석의 예  
Fig. 3. Example of jeongsuk.

## 2. 형세판단 방법

정석을 두는 동안에 형세판단을 하는 방법은 두 가지로 나누어 볼 수 있다. 첫째는, 정석이 한 수 진행되는 상태마다 형세판단하는 경우이다.

둘째는, 정석 하나가 끝날 때마다 형세판단하는 과정을 현재 상태와 관련된 모든 후보 정석에 적용한 결과 가장 유리한 정석을 선택한다.

본 논문에서는 두 번째 방법을 적용하고자 한다. 다음은 이러한 형세판단 과정을 나타내었다.

① 양선수, 편선수, 양후수 순서로 돌과 변사이 경계선과 돌과 돌사이 링크로써 국부적인 영역

을 확정한다.

② 적돌이 존재하지 않는 그룹과 그룹사이, 그룹자체에 대해 돌마다 세력을 계산한다.

③ 모든 흑, 백 그룹에 대해서 ①, ②번을 반복 적용한다.

④ 더이상 그룹이 없으면 정석에 대한 집, 세력을 계산한다.

양선수, 편선수, 양후수에 따라 각각의 형태에 따라 영역을 확정하는 과정을 살펴본다. 편선수의 예로 그림 4에서 a위치처럼 돌과 변사이의 링크가 한 칸 벌림인 경우 백이 선수이면 흑 그룹의 링크는 내부 영역 안쪽으로 백 그룹의 링크는 외부 영역으로 이동하게 된다. 그러므로 흑은 13집, 백은 25집이다.

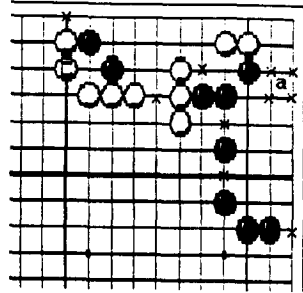


그림 4. 편선수  
Fig. 4. One in which one side has the higher priority.

양선수의 예로 그림 5에서 a위치 처럼 돌과 변사이 링크가 한 칸 벌림이면, 경계선은 한 칸 벌림 형태가 된다. 그러므로 흑은 24집, 백은 16집이다.

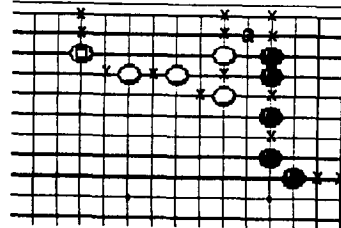


그림 5. 양선수  
Fig. 5. One in which both sides have the same higher priority.

양후수의 예로 그림 6에서 a위치처럼 둘과 번 사이 링크가 한 칸 벌림이면 경계선은 한 칸 벌림 형태가 된다. 그러므로 흑은 16집, 백은 14집이다.

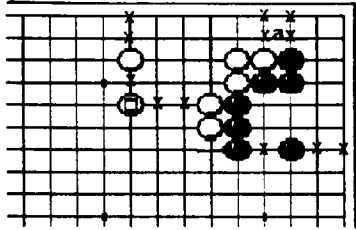


그림 6. 양후수  
Fig. 6. One in which both sides have the same lower priority.

다음 표 1은 그림 4, 그림 5, 그림 6을 전문가가 형세판단한 경우와 본 논문에서 형세판단한 결과를 비교한 것으로서 전문가가 형세판단했을 때 흑과 백의 집 수 차이와 본 논문의 알고리즘을 적용했을 때 흑과 백의 집 수 차이를 비교해 보면 근소한 차이가 있음을 알 수 있다.

표 1. 비교도  
Table 1. Table of comparison.

데이터	구분	전문기사		본 논문	
		흑	백	흑	백
그림 4		13집	22집	13집	25집
그림 5		24집	14집	24집	16집
그림 6		14집	12.5집	16집	14집

### III. 정석 구성/정석 선택 알고리즘

#### 1. 정석 구성 및 진행 방법

정석의 구성은 화점, 소목, 고목, 외목, 삼삼 정석 데이터를 트리로 구성하였으며 그림 7은 저장된 정석 데이터중에서 화점 정석의 일부를 편의상 깊이 3까지만 표현하였다. 트리에서 노드는 전체 5자리로 구성된다. 둘 색깔을 구분하기 위해 1자리, x의 위치값이 2자리, 그리고 y의 위치값이 2자리이다. 예를 들어, 303(백은 0, 흑

은 1이라고 가정)은 x가 3, y가 3인 위치에 있는 백돌을 의미하며 이 상태에서 흑이 선택할 수 있는 정석의 위치는 x가 2, y가 5인 위치에서부터 x가 2, y가 7인 위치까지 5개가 있음을 트리를 통하여 알 수 있다. 바둑이 진행될 때 정석이 적용되어지는 시점에서 이 트리를 활용하고자 한다. 그림 7의 트리에서 깊이가 2일 때 반면에서 나타나는 상황을 그림 8에서 보여주고 있다. 그림 8에서는 백이 화점에 있을 때 흑은 a, b, c, d, e의 위치에 해당하는 정석 데이터에서 하나를 선택하게 된다.

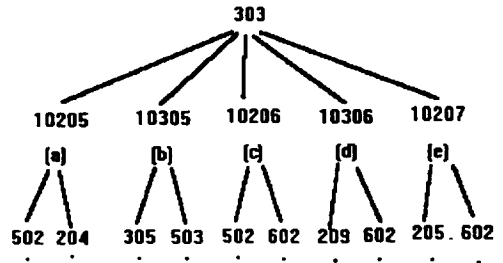


그림 7. 정석 트리  
Fig. 7. Jeongseok tree.

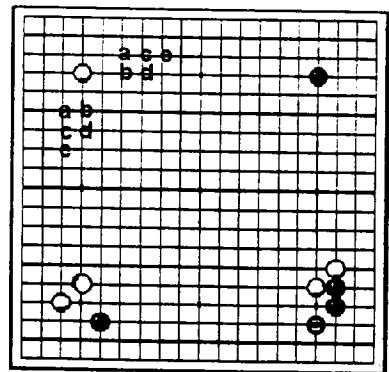


그림 8. 깊이가 2인 반면  
Fig. 8. Board of depth 2.

이러한 정석 데이터를 저장하는 자료구조는 다음과 같다.

form\_type=record

```
position:integer;/*x, y의 위치값*/
sibling: ↑ form_type;/*형제 노드의 위치*/
child: ↑ form_type;/*자식 노드의 위치값*/
endi;
```

정석을 진행하기 위한 자료구조는 다음과 같다.

corner\_type=record

```
no_of_stone:integer;/*돌 갯수*/
first_stone_color:integer;/*첫번째 돌색*/
jungstuk_flag:integer;/*정석인지 check*/
jungstuk_ptr: ↑ form_type;/*정석 pointer*/
endi;
```

no\_of\_stone은 지금까지 놓인 정석의 돌 갯수를 나타내는데 바둑에서 귀는 유리한 위치이므로 먼저 빈귀가 있으면 정석이 수행중이라도 선점하고 빈귀가 없으면 정석이 진행되고 있는 귀를 선택하게 된다.

first\_stone\_color는 정석을 구성하고 있는 첫번째 돌 색깔에 대한 정보를 가지고 있고 jungstuk\_ptr는 가장 최근에 놓여진 수의 다음 위치에 대한 포인터를 기억하고 있다.

이것은 정석이 완전히 끝나지 않은 상태에서 손을 뺀 후 다른 지점에 착수하여 둔후에 돌아와 나머지 정석이 중단되었던 위치부터 계속 수행이 가능하도록 한 것이다. 그리고, 정석이 어느 귀에서 발생했는가를 검사하는 tag를 둠으로써 네귀의 위치값에 맞는 정점으로 변환이 가능하다. 여러곳에서 정석이 진행 중일때는 형세판단 후 가장 좋은 곳을 선택한다.

컴퓨터 바둑에서 정석을 진행하는 과정은 다음과 같이 두 가지의 경우로 나누어 구성된다.

· 컴퓨터 차례인 경우

- ① 빈 귀가 있으면 선택한다.
- ② 빈 귀가 없으면, 정석이 진행중인 귀를 선택한다. (정석이 진행중인 귀가 2개이상이면 형세판단 후 가장 좋은 곳을 선택한다.)
- ③ 일치되는 정석을 수행, 정석 진행을 저장한다.

· 경기자 차례인 경우

- ① 빈 귀, 정석이 진행되고 있는 귀에 놓았을 때, 일치되는 정석을 찾아서 정보를 저장하고 정석 포인터를 이동한다.
- ② 일치되는 정석이 없으면 손을 뺀 경우로서 정석 프로그램을 빠져 나간다.

## 2. 정석 선택 알고리즘

II절에서 형세 판단의 기본적 사항을 설명한 것처럼 먼저 반면에서 흑과 백이 차지하는 영역을 구분 짓는 것이 중요하다. 여기에서 적용되는 우선순위는 양선수, 편선수, 양후수 순서로 하되, 세력도 영역을 구분하는데 있어 중요한 변수로서 사용하였다. 먼저 3가지 형태의 돌과 변사이 경계선과 돌과 돌사이 경계선으로 영역을 구분 지은 후에 그 영역 내부의 집 수를 계산하는 국부적 형세 판단을 실시하게 된다. 반면에 대한 모든 국부적 형세판단이 이루어지면 여기에 세력의 정도를 첨가하여 전체적 형세판단을 하게 된다. 이러한 과정은 임의의 한 정석이 완료된 후에 적용되며 관련되는 정석 트리의 잎 노드마다 실행시킨 결과 값을 평가하여 효율적인 정석을 선택하게 된다.

정석을 선택하는 과정을 좀 더 효율적으로 하기 위해 먼저, 실리적인 측면에서 일부의 정석 데이터를 선택한 후 선택된 정석 데이터중에서 다시 세력의 측면에서 최종적인 정석을 선택하게 된다.

[정석을 선택하는 알고리즘]

- ① 백이 정석을 둘 차례일때,
- ② 더 이상 선택할 정석 데이터가 없을때까지 한개의 정석마다 형세판단 과정을 적용한다.
- ③ 실리적인 관점에서, 유리한 3개의 정석 데이터를 선택한다.
- ④ 세력의 관점에서, ③에서 선택된 3개의 정석 데이터 중에서 가장 유리한 정석을 선택한다.

제시한 알고리즘을 초반 바둑 프로그램에 적용되는 과정을 도식화 하였다.

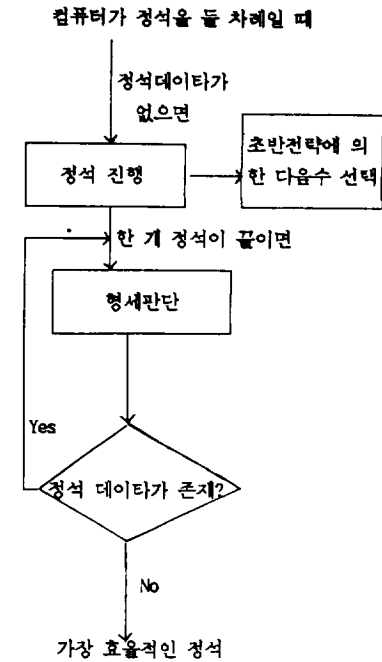


그림 9. 초반 단계  
Fig. 9. Stage of opening state.

IV. 실험 및 고찰

제안한 방법으로 형세판단을 근거로 한 효율적인 정석 선택을 실험하기 위해 월간 바둑<sup>9, 10, 11)</sup>에 실린 10개의 데이터를 실험 데이터로 사용하였으며 그 과정을 그림 10, 그림 11, 그림 12에 각각 나타내었다.

그림 10처럼 우상귀에 흑이 화점, 백이 날일자로 걸친 장면도가 있다. 이와 같은 국면에서 흑이 취할 수 있는 가장 효율적인 정석은 과연 어느 것일까?

그림 11과 같이 트리가 저장되어 있을 때 흑이 선택할 수 있는 정석 데이터는 (a)에서 (f)까지 7개가 있다. 제시한 형세판단 알고리즘을 적용한 결과는 다음과 같다. (a)는 흑 27집, 백 14집이고, (b)는 흑 12집, 백 11집이고, (c)는 흑 27집, 백 14집이

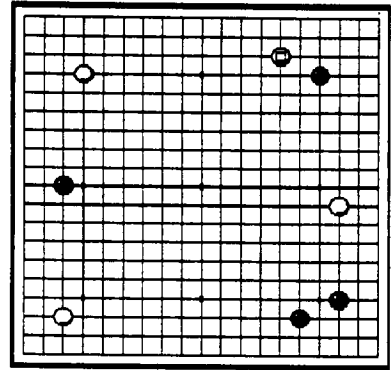


그림 10. 기본도  
Fig. 10. Basic diagram.

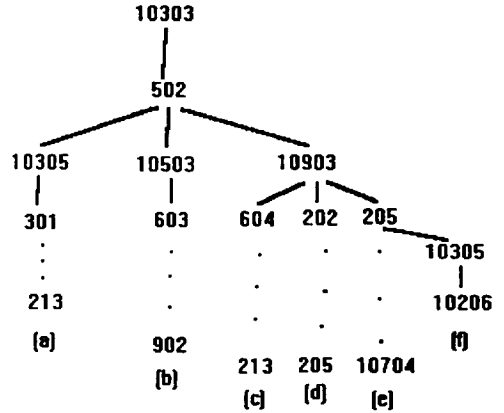


그림 11. 그림 10의 정석 트리  
Fig. 11. Jeongseok tree of fig. 10.

고, (d)는 흑 12집, 백 10집, 흑 세력 5집이고, (e)는 흑 12집, 백 7집, 흑 세력 17집이고, (f)는 흑 16집, 백 10집이 된다. 여기에 정석 선택 알고리즘을 적용한 결과 (c)번 정석 데이터가 가장 효율적임을 알 수 있다. 전문기사는 (c), (f)번 정석 데이터가 그림 10의 상태에서 가장 효율적인 것으로 보였다.

그림 12는 그림 10에서 그림 11의 가장 효율적인 (c)번 데이터를 두었을 때 흑은 우변 17집, 우하귀 10집으로 모두 27집이며 백은 상변 8집, 우변 6집으로 모두 14집이 되는 형세판단 결과

를 보여주고 있다.

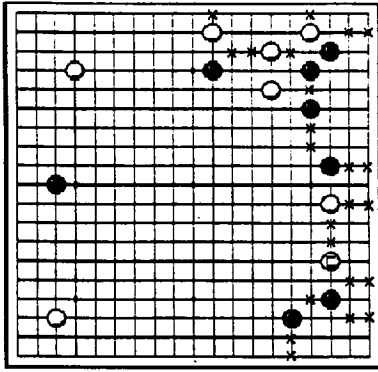


그림 12. (c)를 두었을 때 형세판단  
Fig. 12. State evaluation of (c).

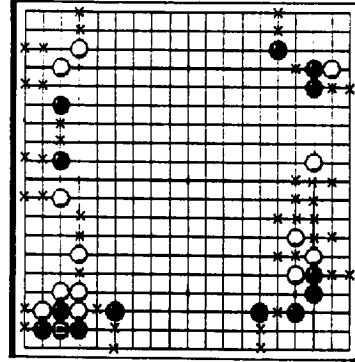
다음 표 2는 본 논문의 알고리즘을 적용한 결과와 전문가의 생각을 비교한 것이다. 첫번째 데이터에서, 2열은 집을 기준으로 선택된 데이터가 3, 4, 6번째임을 의미하며 3열은 세력을 기준으로 최종적으로 선택된 데이터가 6번째임을 나타낸다. 4열은 전문가가 현재 국면에서 가장 효율적이라고 생각한 데이터가 4, 6번째임을 의미하며 5열은 정석을 둘 차례의 둘 색깔을 의미한다. 이 결과 본 논문에서 선택한 데이터와 전문가의 생각이 모두 일치함을 알 수 있다.

표 2. 실험 결과도

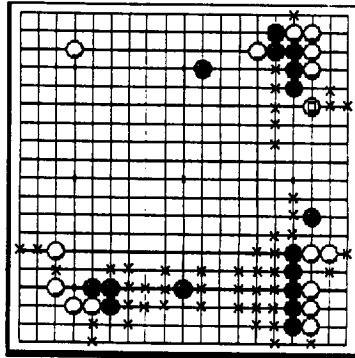
Table 2. Experiment result table.

데이터 구분	본 논문		전문가사	차례
	집	세력		
1	3, 4, 6	6	4, 6	백
2	1, 4, 6	1	1, 6	백
3	2, 3, 6	3	2, 3	백
4	1, 4, 6	4	4, 8	흑
5	3, 4, 9	3	3, 4, 9	흑
6	1, 5, 6	6	6	흑
7	1, 2, 3	2	1, 2, 7	흑
8	2, 5, 6	5	5, 7	백
9	3, 4, 7	3	3, 7	백
10	4, 7, 10	7	7, 10	흑

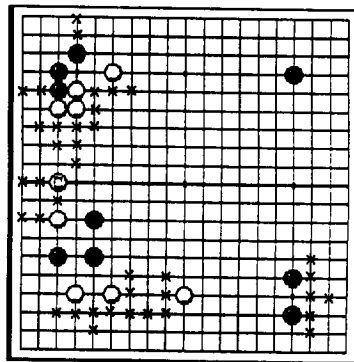
그림13은 표 2에서 선택된 정석을 형세판단한 결과를 나타낸 것으로 1번째에서 3번째 데이터의 결과를 의미한다. 그리고 4번째 데이터의 결과는 그림12에 나타내었다.



1. 흑 42집, 흑 세력 0집, 백 21집, 백 세력 12집



2. 흑 0집, 흑 세력 36집, 백 34집, 백 세력 0집



3. 흑 11집, 흑 세력 6집, 백 2집, 백 세력 24집

그림 13. 표 2의 형세판단  
Fig. 13. State evaluation of table 2.



V. 결 론

본 논문에서는 초반에 형세를 유리하게 하는 전략으로써, 여러 개의 정석 데이터중 현재 국면에서 좀 더 효율적인 정석을 선택하기 위해 형세판단 방법을 적용하였다.

형세판단의 기준으로 집과 세력을 정의하였다. 집은 양선수, 양후수, 편선수를 이용하여 정의하였다.

그룹을 구성하는 돌마다 빈자리에 미치는 영향의 정도를 세력으로 계산하였다.

실험을 통하여 본 논문에서 제시한 알고리즘이 효율적인 정석 선택으로 국면을 유리하게 이끌어 가는 것을 알 수 있었다.

앞으로 연구과제로는 바둑의 진행 과정에서 정확한 형세판단을 하기 위해서는 그룹의 사활판단과 관련되는 지중수에 관한 정의가 더 연구되어야 할 것이다.

참 고 문 헌

1. Bruce Wilcox, "Reflections on building two go programs," *SIGART Newsletter*, Bolt Beranek & Newman Inc., 1985.
2. 김영상, "동적 환경에 적절한 확률론적 형세

- 평가의 모델링에 관한 연구," 석사학위논문, 경북대학교 컴퓨터공학과, 1992.
3. 이동기, "초반 바둑의 계수적 형세 분석," 석사학위논문, 경북대학교 컴퓨터공학과, 1993.
4. 유정혜, "AND/OR 신장 트리를 이용한 바둑 사활 문제 풀이에 관한 연구," 석사학위논문, 경북대학교 컴퓨터공학과, 1991.
5. 최영숙, "순서화된 AND/OR Pruning을 이용한 바둑 사활의 효율 개선," 석사학위논문, 경북대학교 컴퓨터공학과, 1992.
6. 이경자, "후보 선정 알고리즘을 이용한 바둑 끝내기에 관한 연구," 석사학위논문, 경북대학교 컴퓨터공학과, 1992.
7. 박현수, "바둑에서 휴리스틱 함수를 사용한 사활문제 풀이에 관한 연구," 한국정보과학회 학술발표회, vol. 21, no. 1, 1994.
8. 임중권, "바둑에서 후보 선택을 위한 패턴의 계층적 분류와 탐색," 석사학위논문, 경북대학교 컴퓨터공학과, 1993.
9. 양상국, "중반계가법," 월간바둑, 1989. 1 - 1989. 6.
10. 정수현, "형세판단 ABC," 월간바둑, 1992. 7 - 1993. 6.
11. 이창호, "형세판단 XYZ," 월간바둑, 1994. 1 - 8.